BMJ Open

Diagnostic accuracy of X-ray versus CT in COVID-19: a propensity-matched database study

Aditya Borakati ,[1,2] Adrian Perera,[2] James Johnson,[2] Tara Sood[2]

To cite: Borakati A, Perera A, Johnson J, *et al*. Diagnostic accuracy of X-ray versus CT in COVID-19: a propensity-matched database study. *BMJ Open* 2020;**10**:e042946. doi:10.1136/bmjopen-2020-042946

► Prepublication history for this paper is available online. To view these files, please visit the journal online (http://dx.doi.org/10.1136/bmjopen-2020-042946).

Received 20 July 2020
Revised 07 October 2020
Accepted 08 October 2020

[1]Division of Surgery and Interventional Science, University College London, London, UK
[2]Emergency Department, Royal Free Hospital, London, UK

Correspondence to
Dr Aditya Borakati;
a.borakati@doctors.org.uk

ABSTRACT

Objectives To identify the diagnostic accuracy of common imaging modalities, chest X-ray (CXR) and CT, for diagnosis of COVID-19 in the general emergency population in the UK and to find the association between imaging features and outcomes in these patients.

Design Retrospective analysis of electronic patient records.

Setting Tertiary academic health science centre and designated centre for high consequence infectious diseases in London, UK.

Participants 1198 patients who attended the emergency department with paired reverse transcriptase PCR (RT-PCR) swabs for SARS-CoV-2 and CXR between 16 March and 16 April 2020.

Main outcome measures Sensitivity and specificity of CXR and CT for diagnosis of COVID-19 using the British Society of Thoracic Imaging reporting templates. Reference standard was any RT-PCR positive naso-oropharyngeal swab within 30 days of attendance. ORs of CXR in association with vital signs, laboratory values and 30-day outcomes were calculated.

Results Sensitivity and specificity of CXR for COVID-19 diagnosis were 0.56 (95% CI 0.51 to 0.60) and 0.60 (95% CI 0.54 to 0.65), respectively. For CT scans, these were 0.85 (95% CI 0.79 to 0.90) and 0.50 (95% CI 0.41 to 0.60), respectively. This gave a statistically significant mean increase in sensitivity with CT of 29% (95% CI 19% to 38%, p<0.0001) compared with CXR. Specificity was not significantly different between the two modalities. CXR findings were not statistically significantly or clinically meaningfully associated with vital signs, laboratory parameters or 30-day outcomes.

Conclusions Computed tomography has substantially improved diagnostic performance over CXR in COVID-19. CT should be considered in the initial assessment for suspected COVID-19 instead of CXR if capacity allows and balanced against radiation exposure risk.

Strengths and limitations of this study

- Large, appropriately powered, study population consisting of all patients attending the emergency department rather than those solely with confirmed COVID-19; this allowed assessment of specificity for the imaging modalities and applicability to the general population who may attend medical personnel with other complaints, but have underlying SARS-CoV-2 infection.
- Comprehensive statistical analyses were conducted to address confounding in reporting of X-rays including propensity score matching and logistic regression to give a 'doubly robust' model.
- Low amount of missing data and for secondary covariates only; multiple imputation was performed with a good fit, however, observed data would be preferable to imputed data.
- Single centre, retrospective study; potential for inter-reporter and intercentre variability in reporting.
- Large proportion of patients excluded due to not having a reverse transcriptase PCR swab, predominantly, those with imaging reported as negative; this may bias the results towards increased sensitivity and specificity.

INTRODUCTION

SARS-CoV-2 and its resulting disease, COVID-19, have propagated exponentially worldwide, with over 10 million cases in 188 countries at the time of writing.[1 2]

The gold standard for diagnosis of the virus is the detection of viral RNA through reverse transcriptase PCR (RT-PCR) of respiratory tract samples. However, this method has several limitations including (1) low sensitivity at 59%–71%;[3 4] (2) relatively slow turnaround times ranging from a few hours to several days;[5] (3) high expense; and (4) limited capacity for testing in many countries.

CT has been shown to be more sensitive than RT-PCR for diagnosis of COVID-19,[3 4] while being significantly faster and cheaper. This comes with a large radiation dose and capacity is still lacking in many countries.

Plain film chest X-ray (CXR) is ubiquitous worldwide, with a 30–70 times lower dose of radiation[6] and is commonly performed as an initial investigation in COVID-19.

Studies have so far only evaluated imaging in those with confirmed infection; it is, therefore, not possible to calculate the specificity of these modalities. In the context of the global pandemic, infection may be widespread in the community, often with subclinical

Table 1 Ordinal scale used in this study based on the British Society of Thoracic Imaging (BSTI) reporting template[10]

Ordinal scale for study	BSTI grade	Features on X-ray
0	CVCX3-Non-COVID-19	Alternative pathology such as CVCX3—Non-COVID-19 pneumothorax with no features of COVID-19 identified
1	CVCX0—Normal	No pathology seen
2	CVCX2—Indeterminate for COVD-19 or atypical features	Poor quality film or central/basal consolidation
3	CVCX1—Classic findings of COVID-19	Peripheral ground glass opacities

infection.[7 8] A reliable and rapid method to detect infection in the general population, who may present to medical personnel with other complaints, is needed.

Despite its extensive use, the specificity and sensitivity of CXR in the general emergency population for diagnosis of COVID-19 is unknown, nor how imaging features correlate with severity.

This study evaluated the performance of CXR in diagnosing COVID-19 in the emergency department (ED) of a tertiary care hospital.

METHODS

This study was conducted at the Royal Free Hospital, London, UK, an academic health science centre and nationally designated centre for high consequence infectious diseases.[9]

All individuals attending the ED who had paired posterior–anterior chest radiographs and RT-PCR nasopharyngeal swabs for COVID-19 at the time of initial attendance between 16 March 2020 and 16 April 2020 were included.

All chest radiographs were reported by a consultant radiologist and rated on an ordinal scale for probability of COVID-19: alternative pathology identified, not COVID-19; clear chest, unlikely COVID-19; indeterminate findings for COVID-19; classical findings of COVID-19, based on the British Society of Thoracic Imaging's (BSTI) reporting templates (table 1).[10] These were reported prior to RT-PCR results being available.

RT-PCR of swabs were performed in laboratories either at our centre or at a public health laboratory (PHE Collindale, UK), according to published national standard operating procedures.[11] Subsequent RT-PCR swabs taken within 30 days of initial ED attendance were also included.

CT scans performed within 30 days of attendance were retrieved. These were also reported according to the BSTI template. CT pulmonary angiogram was performed in the ED if the D-dimer was >5000 to exclude pulmonary emboli (PE) as per the locally agreed protocol. Subsequent CT chest imaging (whether pulmonary angiogram, contrast or non-contrast) was performed on the basis of clinical suspicion.

Prospectively recorded data were extracted from the Cerner Millennium electronic patient record system (Cerner, Kansas City, Missouri, USA).

Primary outcome

The primary outcome is sensitivity and specificity of initial CXR, where it is reported as having classic COVID-19 features in the ED. This is compared with RT-PCR swab as the reference standard for diagnosis of COVID-19.

In the event of multiple RT-PCR swabs during one attendance, a single positive swab was taken as an overall positive test during one admission.

Secondary outcomes

In those patients who also had CT scans of the thorax, the diagnostic accuracy was compared with CXR, with RT-PCR again as the reference standard. Sensitivity and specificity of CXR when X-rays reported as indeterminate or atypical for COVID-19 were classed as positive was also calculated.

CXR findings were correlated with vital signs at attendance and blood results, including neutrophil counts, D-dimer and C reactive protein, which have been associated with poor prognosis in COVID-19.[12] HRs for clinical outcomes including direct admission to the intensive treatment unit (ITU) from ED and 30-day mortality rates were also calculated for CXR reporting categories.

Statistical analysis

In the event of missing data, multiple imputation was conducted using a Predictive Mean Matching algorithm, via the MICE R package, as described previously.[13] Briefly, this uses a linear regression model (or logistic regression model for categorical data) to find a random value based on already observed data, to replace missing fields.[14] Variables without missing data fields were not modified. The number of imputed datasets was similar in number to the percentage of missing data, as suggested by White and colleagues.[15] Balance diagnostics with density plots are available in online supplemental file 1, adequate balance was assessed via visual inspection of imputed distributions with respect to the original dataset.

The propensity for a CXR being reported as positive or negative for COVID-19 was calculated for several plausible covariates that may influence image characteristics such as age, gender, ethnicity, pre-existing morbidities and the respiratory rate of the patient using a generalised linear model.[16] X-ray positive and negative groups were then matched in each imputed dataset using the nearest neighbour algorithm, with a calliper of 0.2 of the propensity

score SD, without replacement and in random sequential order to obtain a 1:1 match as described elsewhere.[17]

The balance of the match data was assessed quantitatively with mean differences of covariates in each of the X-ray groups before and after matching, with a difference of less than 0.1% considered a good match (online supplementary figures 1 and 2). Visual inspection of matches was also conducted to ensure balance (online supplementary figure 2-4).

After matching, outcome data were adjusted for covariates including age, gender, ethnicity and presence of comorbidities, as well as C reactive protein, D-dimer, troponin and vital signs. This was achieved by generalised linear regression for continuous outcome data, binomial logistic regression for binary categorical outcomes or ordinal logistic regression in the case of CXR where it is the outcome variable.

These regression models were run on each imputed dataset and outcomes were pooled together across each imputed dataset according to Rubin's rules[18] to give an overall estimate.

Diagnostic accuracy statistics

CXRs reported as classical for COVID-19 as per the BSTI guidelines were considered a positive test in the primary analysis. In a secondary analysis, X-rays reported as 'Indeterminate' or 'Atypical' for COVID-19 were also considered positive. All other reports were classified as a negative test. These were compared with nasopharyngeal aspirate RT-PCR results, which were taken as the gold standard for diagnosis of COVID-19. Where more than one swab was taken during the study period (up to 30 days after initial attendance), a single positive result was taken as a positive result for calculation of diagnostic accuracy statistics.

Sensitivity, specificity, predictive values and diagnostic accuracy were calculated using the propensity-matched data after imputation and pooled across imputed datasets with 95% CIs. Apparent and true prevalence based on this dataset are also given for interpretation of the predictive values.

Chest CTs were also reported according to the BSTI guidelines as with X-ray. Diagnostic statistics were calculated on raw, unmatched and non-imputed data (due to a low volume of data for imputation and matching) in the same manner as X-ray. Mean differences and 95% CIs between CT and X-ray for each of the diagnostic statistics are given, with a p value calculated from the CIs.

Agreement between the modalities was assessed on the unmatched dataset, in the sample where CT, CXR and RT-PCR were all available using Cohen's (for two group agreement) and Fleiss' kappa (when all three are compared).

Data presentation

Descriptive statistics are given as means and SD for normally distributed data and as medians and IQRs for non-normally distributed data, before and after matching and multiple imputation (for the latter, these statistics are pooled across imputations).

Association of explanatory variables with SARS-CoV-2 and CXR findings are given as ORs in univariate and multivariate configurations.

Data were considered statistically significant if $p<0.05$. Given the large number of analyses in this paper, data are separately highlighted if $p<0.001$ as a secondary threshold to address the potential for false positives with multiple testing.

Analyses were conducted using R V.4.0.0 (R Foundation for Statistical Computing, Vienna, Austria) and code for the analyses is given in online supplemental file 2.

Sample size calculation

In this study, the lower CI for sensitivity of CXR as reported by Wong *et al*[19] (56%) was used as an estimate of likely sensitivity for COVID-19 . A power of 80% at an alpha of 0.05 was used to calculate the sample size for sensitivity and specificity of 56%. This gave an estimated sample size of 165 in each of the COVID-19 negative and positive groups by RT-PCR (total of 330).

Reporting guidelines

This study is reported according to the STARD guidelines[20] for diagnostic accuracy studies.

Patient and public involvement

Patients and the public were not involved in the design, conduct or dissemination of this study.

RESULTS

A total of 1198 eligible patients with both CXR and RT-PCR were identified in the study period (figure 1). Their characteristics, stratified by positivity for SARS-CoV-2 infection by RT-PCR, are summarised in table 2. This showed that those with confirmed SARS-CoV-2 infection were more likely to be male, older (mean age 66.2 vs 62.7), have lower saturations, higher respiratory rates, while being more likely to be admitted and die within 30 days. There was a significant association with X-ray images and SARS-CoV-2 at baseline, with 59.6% having classic imaging features of COVID-19 in those with positive swabs versus 39.1% in those with negative swabs. There was 8.6% missing data overall in the dataset when variables with >50% missing data were removed and 15 imputations were performed on these remaining variables only.

After multiple imputation for missing data and pooled propensity score matching for plausible covariates that may affect CXR reporting, there were 430 patients in each of the X-ray positive and X-ray negative groups, for a total of 860 patients. Adequate balance was achieved for relevant covariates with a mean difference of <0.1 between groups (online supplemental file 1, table 2).

CT was performed in 302 patients with paired RT-PCR during the same time period, with a median serial interval

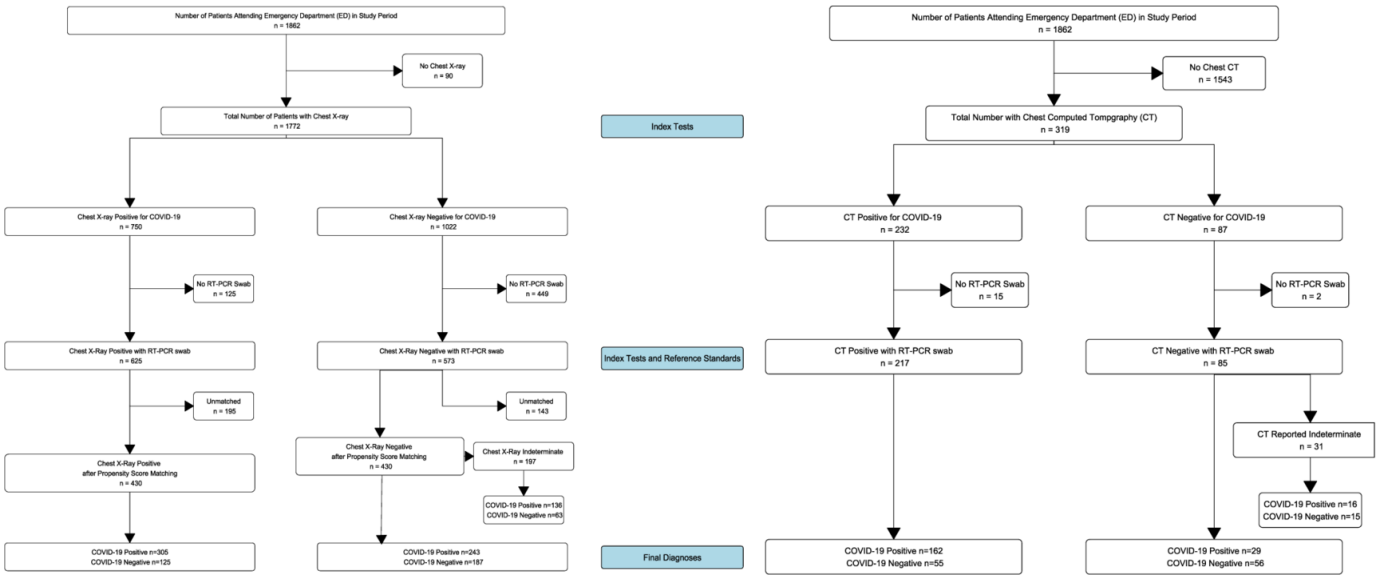


Figure 1 Inclusion and exclusion of patients during study period with test results. RT-PCR, reverse transcriptase PCR.

of 4.5 days (IQR 0–17) after the initial attendance in ED, and of these, 30.1% were within 1 day of attendance.

Diagnostic accuracy

The pooled sensitivity and specificity of CXR was 0.56 (95% CI 0.51 to 0.60) and 0.60 (95% CI 0.54 to 0.65), respectively (table 3). This gave an overall diagnostic accuracy of 0.57 (95% CI 0.54 to 0.61) for CXR.

In comparison, sensitivity and specificity for CT was 0.85 (95% CI 0.79 to 0.90) and 0.50 (95% CI 0.41 to 0.60), respectively. This gave a statistically significant mean increase in sensitivity with CT compared with CXR by 29% (95% CI 19% to 38%, p<0.0001). Specificity was not significantly different between the two modalities. Diagnostic accuracy and negative predictive values were also significantly increased with CT at 0.15 and 0.22, respectively, while the negative likelihood ratio was significantly decreased at −0.44. This shows that the post-test odds of being negative for SARS-CoV-2 by RT-PCR with a negative CT is significantly lower.

Taking X-rays reported as indeterminate as positive increased the sensitivity of CXR to 0.80 (95% CI 0.77 to 0.84), however, specificity was reduced to 0.40 (95% CI 0.35 to 0.46). When CT scans reported as indeterminate are also considered positive, the sensitivity of CT increased to 0.93 (95% CI 0.89 to 0.96), while mean specificity reduced to 0.37 (95% CI 0.28 to 0.47), although this was not statistically different from when indeterminate CTs are considered negative. Sensitivity of CT remained significantly higher than CXR (when indeterminates are considered positive for both) by 0.13 (95% CI 0.05 to 0.19, p<0.001), but specificity was not significantly different between the two.

When comparing only the unimputed, unmatched subset of data where CT, RT-PCR and CXR were all performed (n=287), the agreement between CT and CXR was poor (Cohen's kappa 0.406). Agreement between all three modalities was also poor (Fleiss' kappa 0.361).

Association of CXR with markers of severity and outcomes

Association of covariates with RT-PCR results is shown in table 4 and figure 2. Those who tested positive for SARS-CoV-2 by RT-PCR were significantly more likely to have a classical X-ray (OR 1.79 95% CI 1.25 to 2.56, p<0.002) as would be expected by the diagnostic accuracy statistics (table 4). When the CXR report is considered as an ordered scale, worsening grades of report were associated more strongly with RT-PCR positivity, with a 1.94 times increase in odds for each grade.

Positive CXRs for COVID-19 were significantly associated with lower oxygen saturations (OR 0.94 95% CI 0.92 to 0.97, p<0.001) and temperatures (2.30 95% CI 1.46 to 3.63, p<0.001) in the ED following propensity score matching and multivariate regression (table 5 and figure 3).

They also had higher rates of admission to a general ward from the ED (OR 2.30 95% CI 1.46 to 3.63, p<0.001) but no significant association with 30-day outcomes. There was a statistically significant increase in C reactive protein with a positive X-ray; however, this is unlikely to be clinically meaningful due to the minimal association (OR 1.00 95% CI 1.00 to 1.01).

DISCUSSION

This study is the first to report the diagnostic accuracy of CXR and CT in the general emergency population during the COVID-19 pandemic.

We show that CXR has poor sensitivity and specificity for diagnosis of COVID-19, while CT has 29% higher sensitivity. Many international radiological guidelines advise against CT scanning for the initial assessment of COVID-19[21–23] or where there are equivocal CXRs, while in other countries CT scanning is performed as a routine first-line investigation. Our results suggest that CT should be considered in the initial assessment of COVID-19 and

Table 2 Baseline characteristics of dataset stratified by overall SARS-CoV-2 RT-PCR status, including subsequent swabs during the study period

	SARS-CoV-2 RT-PCR			
	Negative	Positive	P value	Missing (%)
n (%)	435 (36.3)	763 (63.7)		
No of swabs (%)	810 (48.3)	868 (51.7)		
Age (mean (SD))	62.74 (17.72)	66.18 (17.58)	**0.001***	0
Ethnicity			0.097	19
Other Asian (%)	29 (8.0)	72 (11.8)		
South Asian (%)	27 (7.5)	38 (6.2)		
Black (%)	41 (11.4)	91 (14.9)		
Mixed (%)	6 (1.7)	6 (1.0)		
Other (%)	56 (15.5)	105 (17.2)		
White (%)	202 (56.0)	297 (48.8)		
Sex—Male (%)	233 (53.6)	480 (62.9)	**0.002***	0
Oxygen saturation (median (IQR))	95 (6)	93 (8)	**<0.001****	6.3
Respiratory rate (median (IQR))	22 (8)	26 (12)	**<0.001****	6.3
Glasgow Coma Scale (median (IQR))	15 (0)	15 (0)	**0.043***	6.6
Systolic BP (median (IQR))	134 (32)	130 (30)	**0.009***	15.8
Heart rate (median (IQR))	96 (27)	94 (27)	0.092	6.4
Temperature (median (IQR))	37.1 (1.4)	37.7 (1.4)	**<0.001****	6.7
Chest X-ray report			**<0.001****	0
Alternative pathology (%)	4 (0.9)	3 (0.4)		
No abnormalities (%)	178 (4.9)	136 (17.8)		
Indeterminate (%)	83 (19.1)	169 (22.1)		
Classic COVID-19 (%)	170 (39.1)	455 (59.6)		
Presence of comorbidities (%)	297 (79.0)	482 (80.3)	0.669	18.5
Dyspnoea (%)	274 (69.4)	497 (75.5)	0.034	12.1
Neutrophils (median (IQR))	6.42 (4.56)	5.25 (3.92)	**<0.001****	2.3
D-dimer (median (IQR))	1250 (2440)	1105 (1803)	0.204	23.2
Albumin (median (IQR))	39 (7)	37 (6)	**<0.001****	10
C reactive protein (median (IQR))	91.0 (115)	146.5 (264.8)	**<0.001****	3
Creatine kinase (median (IQR))	51 (104)	145 (260)	**<0.001****	23.3
Troponin (median (IQR))	19 (46)	20 (44)	0.278	19.1
Admitted (%)	331 (76.0)	635 (83.2)	**0.003***	0.1
Admitted to ITU (%)	5 (1.3)	32 (4.8)	**0.005***	12.4
30-day follow-up status			**<0.001****	24
Discharged (%)	219 (78.2)	367 (58.3)		
On ambulatory follow-up (%)	14 (5.0)	49 (7.8)		
Admitted (%)	18 (6.4)	60 (9.5)		
Died (%)	29 (10.4)	154 (24.4)		
CT report			**<0.001****	0
No pathology identified (%)	23 (22.1)	6 (3.3)		
Classic COVID-19 findings (%)	52 (50.0)	157 (85.8)		
Indeterminate for COVID-19 (%)	14 (13.5)	14 (7.7)		
Alternative pathology identified (%)	15 (14.4)	6 (3.3)		
Day of symptoms (mean (SD))	9.84 (9.63)	8.56 (15.80)	0.368	69.2

Continued

Table 2 Continued

	SARS-CoV-2 RT-PCR			
	Negative	Positive	P value	Missing (%)

There were 480 additional swabs on 399 unique patients with a median of 2 and mean of 3.5 per patient.
*Significant at p<0.05; **significant at p<0.001.
BP, blood pressure; RT-PCR, reverse transcriptase PCR.

that CXR findings poorly correlate with CT findings in this setting. We also show that indeterminate and non-classical features of COVID-19 significantly increase the sensitivity of these imaging modalities, without a significant decrease in specificity. Further, we demonstrate the limited prognostic value of CXR in COVID-19.

These findings mirror what has previously been reported in the literature on individuals with confirmed COVID-19. Wong *et al*[19] showed a sensitivity of 59% for initial X-ray in confirmed COVID-19 infection, similarly initial case series in China also reported a sensitivity of 59.1%.[12]

A recent press article from Italy reported a much higher sensitivity of 89% for CXR in a smaller general emergency population (n=535) without confirmed COVID-19 at attendance.[24] However, this used telephone follow-up for clinical symptoms of COVID-19 as a reference standard in individuals with an initial negative RT-PCR swab and appeared to classify any abnormal X-ray as positive, which may inflate this figure. When indeterminate CXRs are counted as positive in this study, the sensitivity would be in line with these Italian data. In the USA, a study of patients attending an urgent care centre with confirmed COVID-19 showed a much lower sensitivity at 41.7% for CXR where any abnormality was found on the images.[25] In this study, 97 out of 636 reports were reclassified from 'possible pneumonia' to 'normal' on a second reading from a radiologist, highlighting the importance of inter-rater agreement and possibly explaining this low estimate.

CT has been reported in previous studies as being up to 98% sensitive for the diagnosis of COVID-19 in confirmed patients, when RT-PCR is used as the reference standard in confirmed patients.[3 4] These studies used any potential features of COVID-19 (eg, ground glass opacification, crazy paving) as a positive scan, regardless of spatial distribution or features more characteristic of alternate pathology, unlike the BSTI guidelines used in this study. When we classified indeterminate CTs as positive like these latter studies, our estimates match their sensitivity values.

Consequently, a much lower specificity of 25% was found with initial RT-PCR in the previous literature; however, it is reported that 10 out of 15 (67%) of these negatives subsequently tested positive. This would give an adjusted specificity of 75%, considering subsequent swabs as a reference standard, which, combined with the wider CIs in these smaller studies, would bring estimates in line with the specificity in this paper. More recent meta-analyses have placed the pooled sensitivity of CT in populations with confirmed COVID-19 only, at 89.76% (95% CI 84.42% to 93.84%),[26] in line with the estimates identified here.

Table 3 Diagnostic accuracy metrics for CXR and CT chest with RT-PCR for SARS-CoV-2, as the reference standard

	Chest X-ray	CT chest	Mean difference	P value
Total (n)	860	302		
True positives (n)	305	162	–	–
False positives (n)	125	55	–	–
True negatives (n)	187	56	–	–
False negatives (n)	243	29	–	–
Apparent prevalence (95% CI)	0.50 (0.47 to 0.53)	0.72 (0.66 to 0.77)	**0.22 (0.04 to 0.21)**	**<0.0001****
True prevalence (95% CI)	0.64 (0.60 to 0.67)	0.63 (0.58 to 0.69)	–0.00 (–0.09 to 0.03)	0.111
Sensitivity (95% CI)	0.56 (0.51 to 0.60)	0.85 (0.79 to 0.90)	**0.29 (0.19 to 0.38)**	**<0.0001****
Specificity (95% CI)	0.60 (0.54–0.65)	0.50 (0.41 to 0.60)	–0.10 (–0.25 to 0.04)	0.119
Positive predictive value (95% CI)	0.71 (0.66 to 0.75)	0.75 (0.68 to 0.80)	0.04 (–0.06 to 0.14)	0.492
Negative predictive value (95% CI)	0.43 (0.39 to 0.48)	0.66 (0.55 to 0.76)	**0.22 (0.06 to 0.37)**	**0.005***
Positive likelihood ratio (95% CI)	1.39 (1.19 to 1.62)	1.71 (1.41 to 2.08)	0.32 (–0.22 to 0.89)	0.258
Negative likelihood ratio (95% CI)	0.74 (0.64 to 0.84)	0.30 (0.21 to 0.44)	**–0.44 (–0.64 to –0.21)**	**0.022***
Diagnostic accuracy (95% CI)	0.57 (0.54 to 0.61)	0.72 (0.66 to 0.77)	**0.15 (0.06 to 0.23)**	**<0.0001****

*Significant difference at the <0.05 level; **significant difference at the <0.0001 level.
CXR, chest X-ray; RT-PCR, reverse transcriptase PCR.

Table 4 Association of covariates with RT-PCR status for SARS-CoV-2, following propensity score matching and binomial logistic regression

		SARS-CoV-2 RT-PCR					
		Negative	Positive		OR (univariable)		OR (multivariable)
n		312	548				
Chest X-ray report	Alternative pathology (%)	3 (0.8)	3 (0.5)	–		–	
	No abnormalities (%)	123 (39.6)	104 (19.1)	0.76	(0.08–6.82, p=0.801)	0.48	(0.03–8.82, p=0.620)
	Indeterminate/atypical findings (%)	61 (19.5)	136 (4.8)	1.99	(0.22–17.81, p=0.535)	0.92	(0.05–16.88, p=0.952)
	Classic COVID-19 (%)	125 (40.1)	305 (55.6)	2.17	(0.24–19.19, p=0.484)	1.14	(0.06–20.98, p=0.927)
Age	Mean (SD)	61.8 (17.9)	67.0 (17.7)	**1.02**	**(1.01–1.02, p<0.001)****	**1.02**	**(1.00–1.03, p=0.028)***
Sex	Female (%)	138 (44.3)	212 (38.7)	–		–	
	Male (%)	174 (55.7)	336 (61.3)	1.26	(0.93–1.70, p=0.137)	1.19	(0.83–1.71, p=0.340)
Ethnicity	Other Asian (%)	31 (9.9)	66 (12.0)	–			
	White (%)	164 (52.7)	270 (49.2)	0.76	(0.44–1.31, p=0.326)	0.73	(0.38–1.40, p=0.339)
	Black (%)	39 (12.4)	84 (15.3)	1.01	(0.52–1.98, p=0.974)	0.92	(0.43–1.97, p=0.827)
	Mixed (%)	6 (1.8)	4 (0.8)	0.36	(0.08–1.62, p=0.184)	0.74	(0.11–4.94, p=0.754)
	South Asian (%)	22 (7.0)	36 (6.6)	0.77	(0.34–1.76, p=0.531)	0.68	(0.28–1.65, p=0.390)
	Other (%)	51 (16.2)	89 (16.2)	0.82	(0.43–1.55, p=0.535)	0.88	(0.45–1.74, p=0.716)
Comorbidity	No (%)	65 (20.8)	95 (17.4)	–		–	
	Yes (%)	247 (79.2)	453 (82.6)	1.25	(0.82–1.89, p=0.296)	1	(0.53–1.88, p=0.993)
Dyspnoea on attendance	No (%)	90 (28.8)	139 (25.4)	–		–	
	Yes (%)	222 (71.2)	409 (74.6)	1.19	(0.82–1.73, p=0.356)	0.84	(0.53–1.32, p=0.447)
Oxygen saturation	Median (IQR)	96 (6)	93 (8)	**0.94**	**(0.91–0.97, p<0.001****	0.97	(0.93–1.00, p=0.072)
Respiratory rate	Median (IQR)	23 (8)	25 (8)	**1.04**	**(1.01–1.07, p=0.002)***	1.01	(0.98–1.05, p=0.462)
Glasgow Coma Scale	Median (IQR)	15 (0)	15 (0)	1.02	(0.89–1.17, p=0.819)	1.21	(0.98–1.48, p=0.073)
Temperature	Mean (SD)	37.2 (1.4)	37.7 (1.1)	**1.48**	**(1.26–1.73, p<0.001)****	**1.44**	**(1.20–1.74, p<0.001)****
Heart rate	Mean (SD)	96.7 (20.5)	94.9 (21.5)	1	(0.99–1.00, p=0.305)	1	(0.99–1.01, p=0.702)
Systolic blood pressure	Mean (SD)	136.2 (25.8)	132.6 (24.5)	0.99	(0.99–1.00, p=0.086)	0.99	(0.98–1.00, p=0.097)
Neutrophils	Median (IQR)	6.26 (4.52)	5.05 (3.93)	**0.92**	**(0.89–0.96, p<0.001)****	**0.87**	**(0.82–0.91, p<0.001)****
D-dimer	Median (IQR)	1220 (2343)	1061 (1814)	1	(1.00–1.00, p=0.403)	1	(1.00–1.00, p=0.419)
C reactive protein	Median (IQR)	45 (100)	77 (107)	**1**	**(1.00–1.01, p<0.001)****	**1**	**(1.00–1.01, p=0.021)***
Troponin	Median (IQR)	20 (55)	21 (46)	1	(1.00–1.00, p=0.890)	1	(1.00–1.00, p=0.667)
Albumin	Median (IQR)	39 (7)	37 (6)	0.97	(0.94–1.00, p=0.071)	1.02	(0.98–1.06, p=0.432)
Creatine kinase	Median (IQR)	94 (131)	145 (263)	1	(1.00–1.00, p=0.119)	1	(1.00–1.00, p=0.152)
Admitted from ED	Admitted (%)	235 (75.2)	453 (82.7)	–		–	

Continued

Table 4 Continued

		SARS-CoV-2 RT-PCR			OR (univariable)		OR (multivariable)
		Negative	Positive				
	Discharged (%)	77 (24.8)	95 (17.3)	**1.56**	**(1.06–2.33, p=0.022)****	1.35	(0.79–2.30, p=0.272)
Admitted to ITU from ED	No (%)	307 (98.5)	532 (97.1)	–		–	
	Yes (%)	5 (1.5)	16 (2.9)	1.92	(0.60–6.18, p=0.274)	1.06	(0.25–4.40, p=0.940)
30-day follow-up status	Discharged (%)	259 (83.0)	368 (67.1)	–		–	
	Admitted (%)	22 (6.9)	47 (8.5)	1.53	(0.82–2.87, p=0.181)	1.64	(0.77–3.51, p=0.198)
	Dead (%)	31 (10.1)	133 (24.4)	**3**	**(1.86–4.84, p<0.001)****	**2.81**	**(1.22–6.50, p=0.017)***

*p<0.05; **p<0.001.
ED, emergency department; ITU, intensive treatment unit; RT-PCR, reverse transcriptase PCR.

There is limited coverage in the literature on association of X-ray findings with clinical and laboratory parameters and outcomes in the COVID-19 pandemic. This study demonstrates that classic appearances of COVID-19 were associated with initial lower saturations and lower temperature. Volume opacification of the lung fields was not quantified as a surrogate of severity; however, the use of the BSTI grading templates does this somewhat. When the X-ray report is considered as a graded scale from low likelihood of COVID-19 and severity to high likelihood and severity of disease, there was no significant difference in association with vital signs or laboratory parameters compared with when the X-ray report is merely considered as a binary positive and negative outcome for COVID-19.

Borghesi and colleagues have devised an X-ray grading system, the Brixia score, for severity in admitted patients with confirmed SARS-CoV-2 infection.[27] They further found a significant increase in the severity of CXR by this scoring system in those who were discharged versus those who died.[28 29]

Here, there were no relevant associations between CXR and laboratory values. This analysis also found no association with positive X-rays and 30-day outcomes after multivariate analyses, unlike Borghese *et al*. This is also in contrast to Guan *et al* who found higher rates of ITU admission and death in those with positive imaging findings. However, these studies analysed only those with confirmed SARS-CoV-2 infection. The divergence observed in this study may be due to classifying those with 'Alternate pathology/Indeterminate' or 'CVXC3/CVXC2' as per the BSTI templates, negative for COVID-19 in these analyses. Other studies classified X-rays with any abnormality as a positive for COVID-19. These alternate distributions may still be reflective of underlying COVID-19 and we show significantly higher sensitivity for both CT and CXR when these are classed as positive. It may be that correlating indeterminate X-rays (in addition to classical images) with vitals, laboratory markers and 30-day outcomes would yield significant associations. However this may be unlikely, Xu and Zhang *et al* found that those with classical bilateral and diffuse involvement in upper and lower lobes had more severe disease than those without.[30 31]

There were a total of 70 confirmed PE in our dataset out of 114 CT pulmonary angiograms (61.0%, 5.84% of all patients attending) performed in the ED. The incidence of venous thromboembolism is reported as ranging from 20% to 30% in admitted confirmed SARS-CoV-2 positive patients.[32] Although we have not focused on this cohort of patients in this paper for the sake of brevity and simplicity, this high incidence represents a further advantage for CT over CXR.

CT, even with the absence of contrast, has been shown to have strong accuracy in the diagnosis of PE and many imaging features correlate with the presence of PE. Sensitivities of non-contrast CT for diagnosis of PE have been reported at 96.9% and specificity at 71.9%.[33 34]

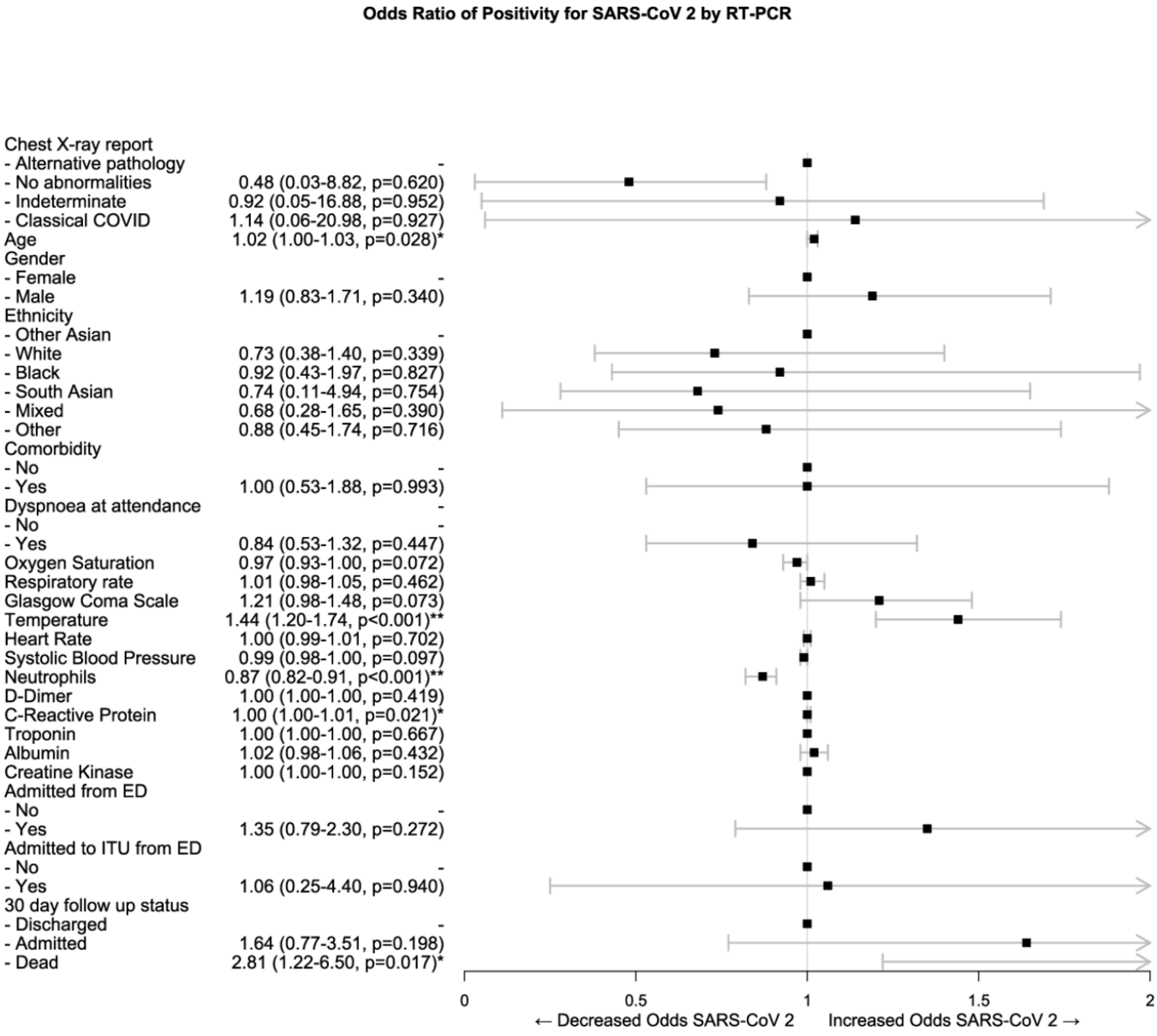


Figure 2 Forest plot of ORs of variables associated with reverse transcriptase PCR positivity for SARS-CoV-2 following multiple imputation, propensity score matching and binomial logistic regression. *Significant difference at the <0.05 level; **significant difference at the <0.001 level. ED, emergency department; ITU, intensive treatment unit.

We therefore see the advantages of CT scanning in COVID-19 as threefold over other diagnostic techniques: (1) the rapid turnaround; (2) increased sensitivity; and (3) the possibility to identify PE in COVID-19, which are a significant burden in this group.

This must be balanced against the excess radiation exposure with CT. Radiation from CT and its association with carcinogenesis is difficult to quantify and no definitive epidemiological studies have confirmed excess risk of cancer.[35] Modern CT scanners and software reconstruction techniques continue to minimise radiation exposure and many ways of shielding parts of the body from radiation also exist. Nevertheless, the excess risk of lifetime cancer is estimated at 1 per 5000 CT examinations.[36]

Strengths and limitations

This study is the largest conducted on imaging during the COVID-19 pandemic and one of the only studies conducted in the general population during the pandemic rather than only in confirmed patients. This enables greater applicability to the clinical setting where the diagnosis is uncertain, in addition to being able to calculate specificity, which is not possible in most studies. This study was planned to be powered to detect a sensitivity and specificity of 56% for CXR and greatly exceeded the sample size necessary for this.

Comprehensive statistical analyses were conducted to account for confounders in both factors influencing reporting of CXR and in factors affecting outcomes. The

Table 5 Association of covariates with CXR report following propensity score matching and either binomial or ordinal logistic regression

		X-ray report							
		Other X-ray	Classical			OR with XR as binary		OR with XR as ordinal	
		Findings	COVID-19	OR (univariable)		Variable (multivariable)		Outcome (multivariable)	
n		430	430						
RT-PCR for SARS-CoV-2	Negative (%)	187 (43.4)	125 (29.1)	–		–		–	
	Positive (%)	243 (56.6)	305 (7.9)	**1.85**	**(1.36–2.56, p<0.001)****	**1.79**	**(1.25–2.56, p<0.002)***	**1.94**	**(1.37–2.76, p<0.001)****
Age	Mean (SD)	65.0 (18.9)	65.3 (16.9)	1	(0.99–1.01, p=0.849)	0.99	(0.98–1.00, p=0.164)	1	(0.99–1.01, p=0.542)
Sex	Female (%)	176 (40.9)	175 (40.6)	–		–			
	Male (%)	254 (59.1)	255 (59.3)	1.01	(0.75–1.37, p=0.940)	0.87	(0.63–1.20, p=0.400)	1.02	(0.49–2.09, p=0.967)
Ethnicity	Other Asian (%)	49 (11.4)	48 (11.2)	–		–			
	South Asian (%)	29 (6.7)	29 (6.7)	1.04	(0.52–2.04, p=0.912)	1.02	(0.47–2.17, p=0.965)	1.02	(0.49–2.09, p=0.967)
	Black (%)	61 (14.2)	61 (14.2)	1.02	(0.55–1.85, p=0.957)	0.88	(0.46–1.69, p=0.719)	0.92	(0.52–1.65, p=0.789)
	Mixed (%)	5 (1.2)	5 (1.2)	0.92	(0.21–4.00, p=0.911)	0.86	(0.18–4.17, p=0.853)	0.85	(0.17–4.30, p=0.838)
	Other (%)	70 (16.3)	70 (16.3)	1.02	(0.58–1.79, p=0.943)	0.98	(0.52–1.82, p=0.942)	0.93	(0.53–1.64, p=0.810)
	White (%)	216 (50.2)	217 (50.5)	1.03	(0.63–1.67, p=0.913)	0.97	(0.57–1.67, p=0.926)	0.9	(0.55–1.47, p=0.666)
Comorbidity	No (%)	82 (19.1)	78 (18.1)	–		–			
	Yes (%)	348 (80.9)	352 (81.9)	0.95	(0.66–1.36, p=0.777)	0.93	(0.59–1.49, p=0.782)	0.88	(0.57–1.37, p=0.592)
Dyspnoea	No (%)	191 (29.3)	103 (24.0)	–		–			
	Yes (%)	304 (70.7)	327 (76.0)	1.31	(0.92–1.88, p=0.123)	1.2	(0.80–1.82, p=0.380)	1.22	(0.83–1.80, p=0.301)
Oxygen saturation	Median (IQR)	95 (7)	93 (7)	**0.94**	**(0.91–0.96, p<0.001)****	**0.94**	**(0.92–0.97, p<0.001)****	**0.94**	**(0.91–0.97, p<0.001)****
Respiratory rate	Median (IQR)	24 (10)	24 (10)	1.01	(0.99–1.02, p=0.570)	0.97	(0.94–1.00, p=0.063)	0.98	(0.96–1.01, p=0.157)
Glasgow Coma Scale	Median (IQR)	15 (0)	15 (0)	1.04	(0.92–1.19, p=0.524)	1.05	(0.90–1.23, p=0.503)	1.05	(0.92–1.21, p=0.464)
Temperature	Mean (SD)	37.6 (1.1)	37.5 (1.3)	0.93	(0.83–1.06, p=0.297)	**0.79**	**(0.67–0.93, p=0.006)***	**0.85**	**(0.73–0.99, p=0.031)***

Continued

Table 5 Continued

		X-ray report							
		Other X-ray	Classical			OR with XR as binary		OR with XR as ordinal	
		Findings	COVID-19	OR (univariable)		Variable (multivariable)		Outcome (multivariable)	
Heart rate	Mean (SD)	95.7 (21.4)	95.5 (21.0)	1	(0.99–1.01, p=0.888)	1	(0.99–1.01, p=0.864)	1	(0.99–1.01, p=0.872)
Systolic blood pressure	Mean (SD)	133.8 (25.0)	134.0 (25.6)	1	(0.99–1.01, p=0.907)	1	(0.99–1.01, p=0.335)	1	(1.00–1.01, p=0.478)
Neutrophils	Median (IQR)	5.44 (4.54)	5.67 (4.03)	1	(0.97–1.04, p=0.892)	0.96	(0.92–1.01, p=0.143)	0.96	(0.92–1.01, p=0.115)
D-dimer	Median (IQR)	1119 (2221)	1119 (1850)	1	(1.00–1.00, p=0.513)	1	(1.00–1.00, p=0.568)	1	(1.00–1.00, p=0.385)
C reactive protein	Median (IQR)	46 (93)	88 (110)	**1**	**(0.99–1.00, p<0.001)****	**1**	**(1.00–1.01, p<0.001)****	**1**	**(1.00–1.01, p<0.001)****
Troponin	Median (IQR)	23 (54)	20 (46)	1	(1.00–1.00, p=0.231)	1	(1.00–1.00, p=0.277)	1	(1.00–1.00, p=0.059)
Albumin	Median (IQR)	39 (7)	37 (6)	**0.93**	**(0.90–0.96, p<0.001)****	**0.93**	**(0.90–0.97, p=0.001)***	**0.94**	**(0.91–0.97, p=0.001)***
Creatine kinase	Median (IQR)	110 (183)	134 (239)	1	(1.00–1.00, p=0.535)	1	(1.00–1.00, p=0.242)	1	(1.00–1.00, p=0.186)
Admitted from ED	Admitted (%)	315 (73.3)	373 (86.7)	**2.37**	**(1.63–3.46, p<0.001)****	**2.3**	**(1.46–3.63, p<0.001)****	**2.22**	**(1.47–3.33, p<0.001)****
	Discharged (%)	115 (26.7)	57 (13.3)	–		–		–	
Admitted to ITU from ED	No (%)	423 (98.4)	416 (96.7)	–		–			
	Yes (%)	7 (1.6)	14 (3.3)	2.17	(0.69–6.67, p=0.181)	1.27	(0.32–5.00, p=0.732)	1.34	(0.36–5.00, p=0.653)
30-day follow-up status	Discharged (%)	316 (73.5)	311 (72.3)	–		–			
	Admitted (%)	34 (7.9)	34 (7.9)	1.31	(0.81–2.13, p=0.282)	1.32	(0.69–2.53, p=0.392)	1.43	(0.78–2.63, p=0.653)
	Dead (%)	80 (18.6)	85 (19.8)	1.03	(0.73–1.45, p=0.886)	1.38	(0.80–2.37, p=0.247)	1.41	(0.87–2.27, p=0.157)

*p<0.001.
ED, emergency department ; ITU, intensive treatment unit; RT-PCR, reverse transcriptase PCR .

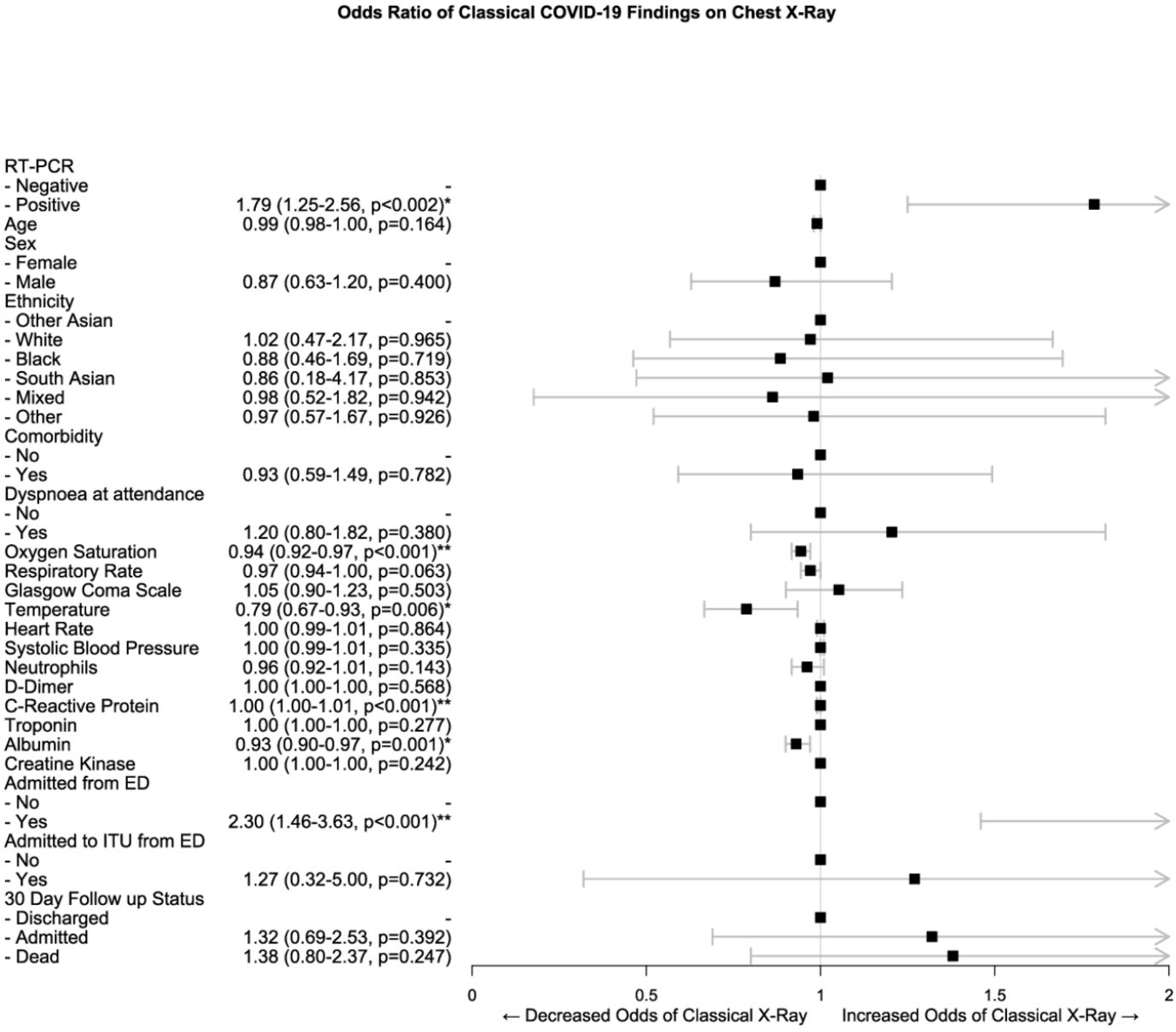


Figure 3 Forest plot of ORs of variables associated with classical chest X-ray features COVID-19 following propensity score matching and binomial logistic regression. *Significant difference at the <0.05 level; **significant difference at the <0.001 level. ED, emergency department; ITU, intensive treatment unit; RT-PCR, reverse transcriptase PCR.

data were collected from prospectively maintained electronic records; however, the retrieval took place retrospectively with its inherent disadvantages. We were not able to collect data on several relevant covariates such as specific comorbidities or markers of severity such as lymphocytes. Furthermore, there was a significant amount of missing data that required multiple imputation to replace, although the fit of these imputed data was good, actual, observed data would be ideal.

Inter-rater reliability of imaging reports was not analysed in this paper and there was the potential for individual radiologists to have greater or lesser accuracy in the diagnosis of COVID-19.The literature has so far suggested a strong degree of agreement between radiologists in reporting of COVID-19 images.[28]

The single centre nature of this study further limits generalisability and the potential for interhospital disagreement in imaging, in addition to inter-rater disagreement.

Finally, the median time for patients to receive a CT scan was 4.5 days following initial attendance to ED. Thus, the scans may not have been directly comparable to the initial CXR, both because of the progression of disease and because the SARS-CoV-2 status may have been confirmed at this point, biassing the reporting of these scans.

Future research

Although this study used RT-PCR of nasopharyngeal swabs as a reference standard, newer methods exist for

diagnosis of the disease. Serological assays for antibodies against SARS-CoV-2 are increasingly available and may represent a better gold standard in diagnosis for future research.[37] RT-PCR is limited by swabbing technique for nasopharyngeal samples and the fact that the virus is more avid in the lower respiratory tract.[38] However, many patients may not seroconvert prior to death, limiting this test to survivors only.

Point-of-care lung ultrasound is a new technique for diagnosis of COVID-19 which may mitigate many of the issues noted with the modalities discussed so far. It has no radiation, is fast, cheap and may be able to detect lower respiratory tract disease unlike nasopharyngeal swab.

However, there is limited evidence beyond small case series on its diagnostic accuracy.[39–41] Further, like other ultrasound techniques, accuracy will likely be operator dependent[42] and experience will need to be built up for robust results in evaluating suspected COVID-19.

Finally, much research has been conducted in the use of artificial intelligence techniques to correctly diagnose COVID-19 based on imaging.[43–45] These techniques would obviate capacity limitations in reporting imaging as well as eliminate inter-reporter variability. However, as with any supervised machine learning technique, large, generalisable datasets, with correctly preclassified positive and negative cases (which in turn will depend on a truly accurate reference standard), are needed.[46]

CONCLUSION

CXR has poor sensitivity and specificity in diagnosing COVID-19 in the general population during the pandemic. CT scanning has demonstrated excellent sensitivity and should strongly be considered during the pandemic in the initial assessment of COVID-19. This needs to be balanced against the risk of excess radiation with CT, where capacity allows.

Correction notice This article has been corrected since it was first published. Counclusion section of abstract has been corrected.

Acknowledgements We would like to thank Scott Wilson from the Royal Free Hospital's clinical practice group analytics department for retrieving the data from the hospital's data warehouse. We would like to thank Dr Federico Ricciardi of the Department of Statistical Science and PRIMENT Clinical Trials Unit at University College London for reviewing the statistical methods in this study.

Funding The authors have not declared a specific grant for this research from any funding agency in the public, commercial or not-for-profit sectors.

Competing interests None declared.

Patient consent for publication Not required.

Ethics approval This study was registered with the local institutional review board as a service evaluation using anonymised data only. No formal ethics committee review was required.

Provenance and peer review Not commissioned; externally peer reviewed.

Data availability statement Anonymised data is available on reasonable request from the corresponding author. Analysis scripts are attached as a supplementary file.

ORCID iD

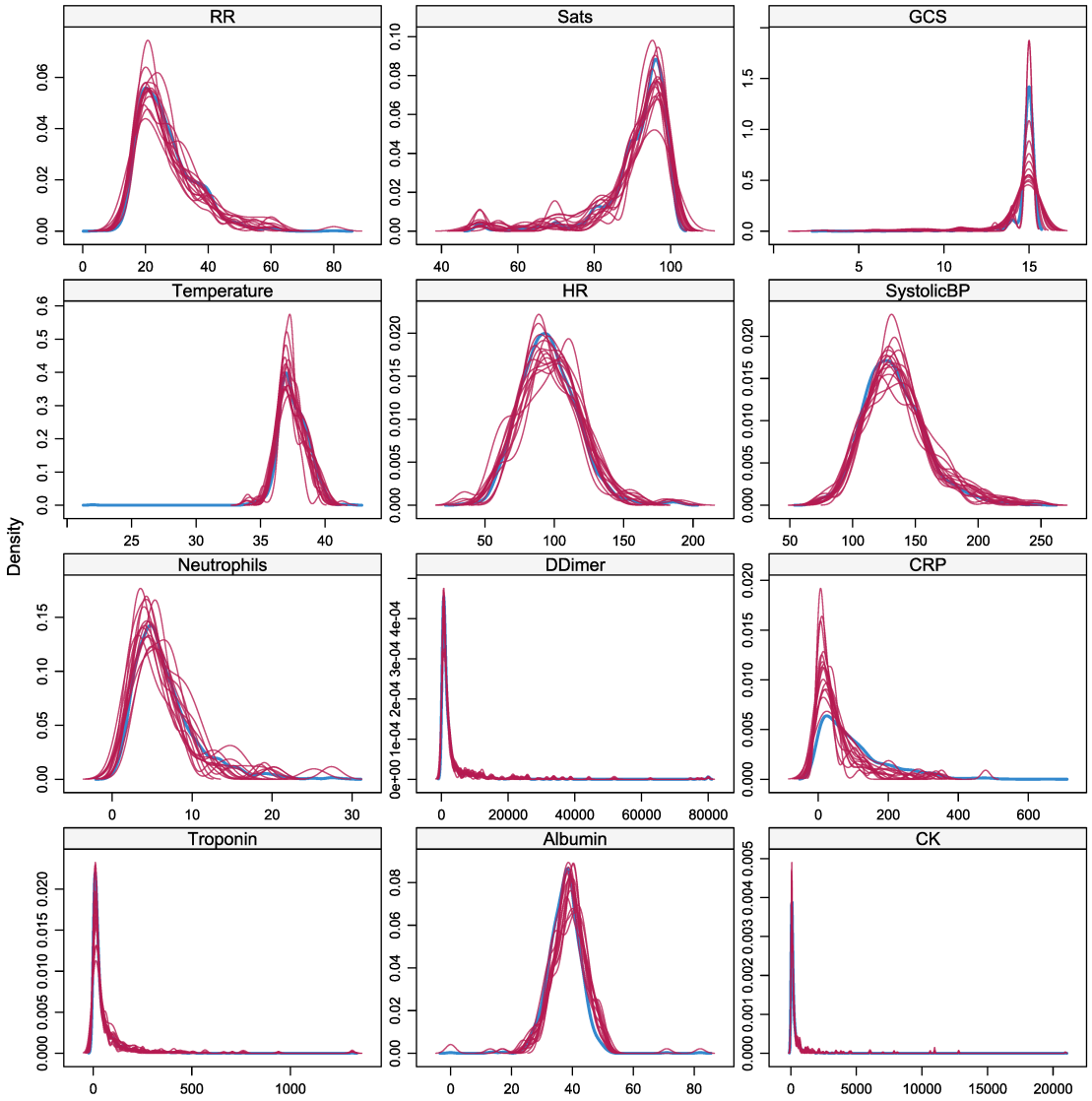
Aditya Borakati http://orcid.org/0000-0003-0457-4944

REFERENCES

1 COVID-19 map. Johns Hopkins coronavirus Resour. *Cent* https://coronavirus.jhu.edu/map.html (accessed 30 Jun 2020).
2 Dong E, Du H, Gardner L. An interactive web-based dashboard to track COVID-19 in real time. *Lancet Infect Dis* 2020;0.
3 Ai T, Yang Z, Hou H, *et al*. Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. *Radiology* 2020;296:E32–40.
4 Fang Y, Zhang H, Xie J, *et al*. Sensitivity of chest CT for COVID-19: comparison to RT-PCR. *Radiology* 2020;296:E115–7.
5 Konrad R, Eberle U, Dangel A, *et al*. Rapid establishment of laboratory diagnostics for the novel coronavirus SARS-CoV-2 in Bavaria, Germany, February 2020. *Eurosurveillance* 2020;25.
6 Lin EC. Radiation risk from medical imaging. *Mayo Clin Proc* 2010;85:1142–6.
7 Mizumoto K, Kagaya K, Zarebski A, *et al*. Estimating the asymptomatic proportion of coronavirus disease 2019 (COVID-19) cases on board the diamond Princess cruise SHIP, Yokohama, Japan, 2020. *Eurosurveillance* 2020;25:2000180.
8 Arons MM, Hatfield KM, Reddy SC, *et al*. Presymptomatic SARS-CoV-2 infections and transmission in a skilled nursing facility. *N Engl J Med* 2020;382:2081–90.
9 GOV.UK. High consequence infectious diseases (HCID). Available: https://www.gov.uk/guidance/high-consequence-infectious-diseases-hcid [Accessed 24 May 2020].
10 Desai S. COVID-19 BSTI reporting templates | the British Society of thoracic imaging. *Br. Soc. Thorac. Imaging* 2020 https://www.bsti.org.uk/covid-19-resources/covid-19-bsti-reporting-templates/ (accessed 29 Apr 2020).
11 NHS England. Guidance and standard operating procedure: COVID-19 virus testing in NHS laboratories, 2020. Available: https://www.england.nhs.uk/coronavirus/wp-content/uploads/sites/52/2020/03/guidance-and-sop-covid-19-virus-testing-in-nhs-laboratories-v1.pdf [Accessed 24 May 2020].
12 Guan W, Ni Z, Hu Y, *et al*. Clinical characteristics of coronavirus disease 2019 in China. *N Engl J Med* 2020;0:null.10/ggm6dh.
13 Honaker J, King G, Blackwell M. Amelia II: A Program for Missing Data. *J Stat Softw* 2011;45.
14 van Ginkel JR, Linting M, Rippe RCA, *et al*. Rebutting existing misconceptions about multiple imputation as a method for handling missing data. *J Pers Assess* 2020;102:297–308.
16 White IR, Royston P, Wood AM. Multiple imputation using chained equations: issues and guidance for practice . *Stat Med 2011;30:377–99.10.1002/sim.4067*. 15.
16 He H, McDermott MP. A robust method using propensity score stratification for correcting verification bias for binary tests. Biostat Oxf Engl 2012;13:32–47.10/c4jzn6.
17 DE H, Imai K, King G, *et al*. MatchIt : Nonparametric Preprocessing for Parametric Causal Inference. *J Stat Softw* 2011;42.
18 Marshall A, Altman DG, Holder RL, *et al*. Combining estimates of interest in prognostic modelling studies after multiple imputation: current practice and guidelines. *BMC Med Res Methodol* 2009;9:57.
19 Wong HYF, HYS L, Fong AH-T, *et al*. Frequency and distribution of chest radiographic findings in COVID-19 positive patients. *Radiology*2020:201160.
20 Cohen JF, Korevaar DA, Altman DG, *et al*. Stard 2015 guidelines for reporting diagnostic accuracy studies: explanation and elaboration. *BMJ Open* 2016;6:e012799.
21 Rubin GD, Ryerson CJ, Haramati LB, *et al*. The role of chest imaging in patient management during the COVID-19 pandemic:

a multinational consensus statement from the Fleischner Society. *Radiology*2020:201365.
22 ACR recommendations for the use of chest radiography and computed tomography (CT) for suspected COVID-19 infection. Available: https://www.acr.org/Advocacy-and-Economics/ACR-Position-Statements/Recommendations-for-Chest-Radiography-and-CT-for-Suspected-COVID19-Infection [Accessed 5 Jun 2020].
23 British Society of Thoracic Imaging. COVID-19: BSTI statement and guidance, 20201. Available: https://www.bsti.org.uk/media/resources/files/COVID11.3.20_2.pdf [Accessed 5 Jun 2020].
24 Schiaffino S, Tritella S, Cozzi A, *et al*. Diagnostic performance of chest X-ray for COVID-19 pneumonia during the SARS-CoV-2 pandemic in Lombardy, Italy. *J Thorac Imaging* 2020.
25 Weinstock MB, Echenique A, Russell JW, *et al*. *Chest X-ray findings in 636 ambulatory patients with COVID-19*. 10. Presenting to an Urgent Care Center: A Normal Chest X-Ray Is no Guarantee.
26 Bao C, Liu X, Zhang H, *et al*. Coronavirus disease 2019 (COVID-19) CT findings: a systematic review and meta-analysis. *Journal of the American College of Radiology* 2020;17:701–9.
27 Borghesi A, Zigliani A, Masciullo R, *et al*. Radiographic severity index in COVID-19 pneumonia: relationship to age and sex in 783 Italian patients. *Radiol Med* 2020;125:461–4.
28 Borghesi A, Maroldi R. COVID-19 outbreak in Italy: experimental chest X-ray scoring system for quantifying and monitoring disease progression. *Radiol Med* 2020;125:509–13.
29 Borghesi A, Zigliani A, Golemi S, *et al*. Chest X-ray severity index as a predictor of in-hospital mortality in coronavirus disease 2019: a study of 302 patients from Italy. *International Journal of Infectious Diseases* 2020;96:291–3.
30 Y-H X, Dong J-H, W-M A, *et al*. Clinical and computed tomographic imaging features of novel coronavirus pneumonia caused by SARS-CoV-2. *J Infect* 2020;80:394–400.
31 Zhang J-J, Dong X, Cao Y-Y, *et al*. Clinical characteristics of 140 patients infected with SARS-CoV-2 in Wuhan, China. allergy (Published Online First: 19 February 2020).
32 Lodigiani C, Iapichino G, Carenzo L, *et al*. Venous and arterial thromboembolic complications in COVID-19 patients admitted to an academic hospital in Milan, Italy. *Thromb Res* 2020;191:9–14.
33 Chien C-H, Shih F-C, Chen C-Y, *et al*. Unenhanced multidetector computed tomography findings in acute central pulmonary embolism. *BMC Med Imaging* 2019;19:65.
34 Mohamed N, Othman MM, Hassan L, *et al*. The accuracy of non-contrast chest computed tomographic scan in the detection of pulmonary thromboembolism. *J Curr Med Res Pract* 2019;4:61.
35 McCollough CH, Bushberg JT, Fletcher JG, *et al*. Answers to common questions about the use and safety of CT scans. *Mayo Clinic Proceedings* 2015;90:1380–92.
36 Moser JB, Sheard SL, Edyvean S, *et al*. Radiation dose-reduction strategies in thoracic CT. *Clin Radiol* 2017;72:407–20.
37 Long Q-X, Liu B-Z, Deng H-J, *et al*. Antibody responses to SARS-CoV-2 in patients with COVID-19. *Nat Med* 2020:1–4.
38 Wang W, Xu Y, Gao R, *et al*. Detection of SARS-CoV-2 in different types of clinical specimens. *JAMA* 2020;323:1843–4.
39 Smith MJ, Hayward SA, Innes SM, *et al*. Point-of-care lung ultrasound in patients with COVID-19 - a narrative review. *Anaesthesia* 2020;75:1096–1041.
40 Haaksma ME, Heldeweg MLA, Matta JEL, *et al*. Lung ultrasound findings in patients with novel SARS-CoV2. *medRxiv*2020;2020:05.18.20105775.
41 Benchoufi M, Bokobza J, Chauvin AA, *et al*. Lung injury in patients with or suspected COVID-19 : a comparison between lung ultrasound and chest CT-scanner severity assessments, an observational study. *medRxiv*2020;2020:04.24.20069633.
42 Fine D, Perring S, Herbetko J, *et al*. Three-Dimensional (3D) ultrasound imaging of the gallbladder and dilated biliary tree: reconstruction from real-time B-scans. *Br J Radiol* 1991;64:1056–7.
43 Shi F, Wang J, Shi J, *et al*. Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19. *IEEE Rev Biomed Eng* 2020:1.
44 Li L, Qin L, Xu Z, *et al*. Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. radiology 2020;:200905.10/ggpdgp.
45 Wang L, Wong A. COVID-Net: a tailored deep Convolutional neural network design for detection of COVID-19 cases from chest X-ray images. ArXiv200309871 Cs Eess. Available: http://arxiv.org/abs/2003.09871 [Accessed 13 Jun 2020].
46 Kotsiantis SB. Use of machine learning techniques for educational proposes: a decision support system for forecasting students' grades. *Artif Intell Rev 2012;37:331–44.10/fmbng4 19*.

Supplementary file 1



Supplementary figure 1- Density plots of imputed datasets; Blue represents original dataset; other colours are individual imputed datasets (n=15)

Covariate:	Means Treated	Means Control	Standard Deviation Control	Mean Difference
Overall Propensity Score	0.422997940	0.53935303	0.1449627	-0.1163550897
Female	36.3782051	45.026178	0.4979547	-8.64797288
Male	63.6217949	54.973822	0.4979547	8.64797288
Age	63.796474359	66.19022688	18.5893357	-23.937525171
Comorbidity- Yes	76.1217949	84.467714	0.3625287	-8.34591892
Ethnicity- South Asian	6.5705128	6.631763	0.2490539	-0.06124983
Ethnicity- Black	16.1858974	11.518325	0.3195219	4.66757283
Ethnicity- Mixed	0.9615385	1.396161	0.1174340	-0.43462210
Ethnicity- Other	18.9102564	13.263525	0.3394765	5.64673110
Ethnicity- White	46.6346154	57.766143	0.4943635	-11.13152772
Respiratory Rate	29.214743590	24.01745201	7.2639816	5.1972915828

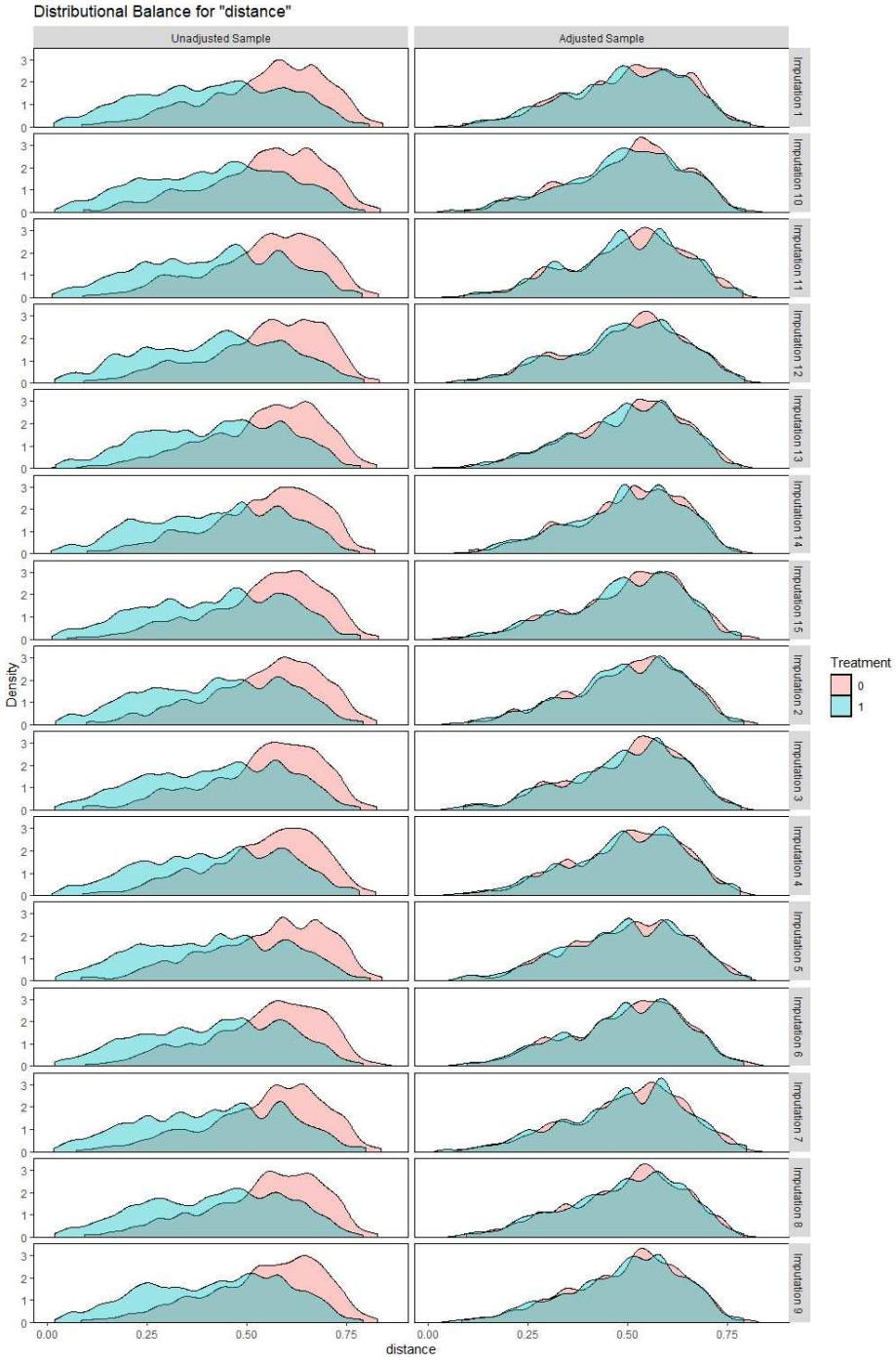
Supplementary table 1- Means of data before multiple imputation and propensity score matching

	Type	Minimum Difference Adjusted	Mean Difference Adjusted	Maximum Difference Adjusted
Distance	Distance	0.016988	0.027107	0.040963
Sex = Male	Binary	-0.03917	-0.0028	0.015982
Age	Contin.	-0.04586	-0.01371	0.027589
Comorbidity = Yes	Binary	-0.02331	-0.00778	0.004598
Ethnicity = Other Asian	Binary	-0.01392	0.002362	0.016471
Ethnicity = South Asian	Binary	-0.01399	-0.00136	0.011905
Ethnicity = Black	Binary	-0.01852	0.000443	0.015982
Ethnicity = Mixed	Binary	-0.00464	0.001403	0.007042
Ethnicity = Other	Binary	-0.01152	4.30E-06	0.00939
Ethnicity = White	Binary	-0.02353	-0.00285	0.018433
Respiratory Rate	Contin.	-0.06157	-0.03478	-0.00442

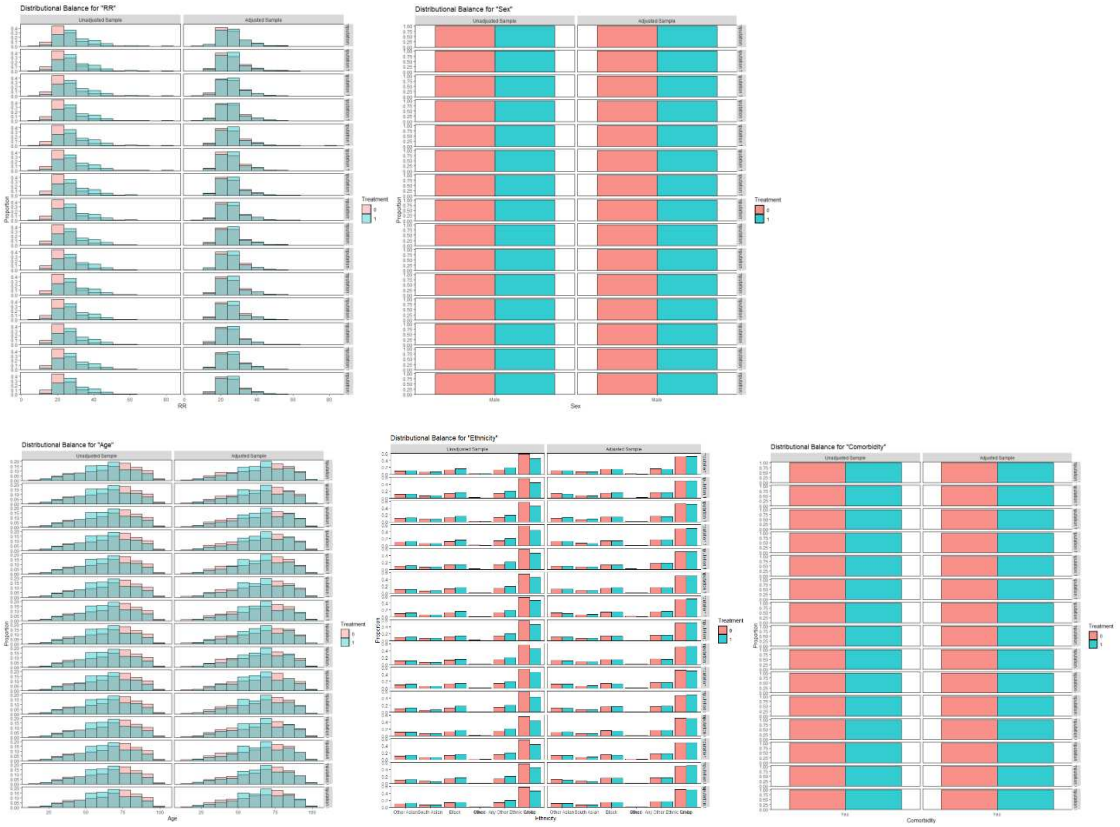
Supplementary table 2- Balance summary across imputations

	XR- Negative	XR- Positive	Total
All	573	625	1,198
Matched	430	430	860
Unmatched	143	195	338
Discarded	0	0	0

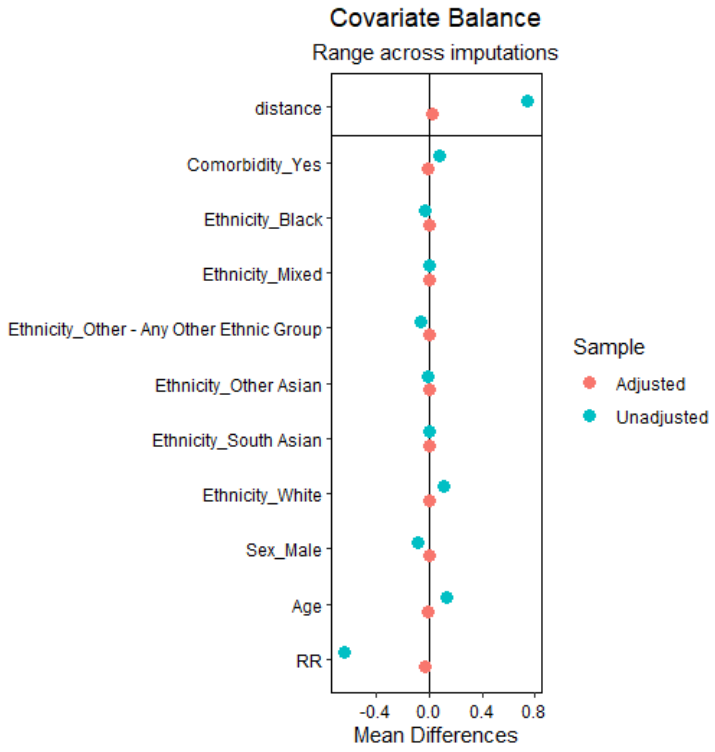
Supplementary table 3- Average Sample sizes pre- and post- matching across imputed data sets



Supplementary figure 2- Density plot of propensity scores pre- and post- matching in each imputed dataset; treatment units represent a positive X-ray for COVID-19, whereas a control unit represents a negative X-ray



Supplementary figure 3- Histogram of distributions for each matching covariate pre- and post- matching in each imputed dataset; treatment units represent a positive X-ray for COVID-19, whereas a control unit represents a negative X-ray



Supplementary figure 4- Love plot of pooled balances across imputed datasets in matching covariates after matching

CXR in COVID Analysis

Dr Aditya Borakati

Royal Free Hospital, Pond Street, London, NW3 2QG
a.borakati@doctors.org.uk

2020-10-06

/

1 Software Environment and Packages 7
1.1 Load Packages and Data 9
1.1.1 Load Packages: 9
1.2 Power Calculation 9
2 Load Data: 11
3 Data Cleaning 13
3.0.1 Follow Up Swabs + Initial Swabs Positive: .14
3.0.2 Paired XR and RT-PCR data 14
4 Demographic table of raw data 17
5 Imputation 21
6 Propensity Score Matching 23
6.1 Match Balance Diagnostics 23
7 Matched Demographics Table: 25
8 Diagnostic Accuracy 27
8.1 CT Data and Accuracy 28
8.2 CT and XR accuracy comparison 30
8.2.1 Sensitivity 30
8.3 Intermodality Agreement 34
8.3.1 Diagnostic Accuracy Analysis when Indeterminate Reports of CXR and CT are taken as positive 35
9 Pooled Regression after Multiple Imputation and Propensity Score Matching 39
9.0.1 Pooled Univariate Odds Ratios for OverallPos as dependent variable 39
9.0.2 Binomial Logistic Regression with Positive Chest X-ray Report as Dependent Variable 40
9.0.3 Univariate XRPositive as dependent 40
9.0.4 Multivariate XRPositive as dependent 40
9.0.5 Pooled Ordinal Logistic Regression with XRPositive as dependent 41

9.1 Forest Plots 41
9.2 Correlation Matrix 43
9.3 STARD Flow Diagram 46

1 Software Environment and Packages

```
R version 4.0.0 (2020-04-24)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 19041)
Matrix products: default
locale:
LC_COLLATE=English_United Kingdom.1252  LC_CTYPE=English_United Kingdom.1252
LC_MONETARY=English_United Kingdom.1252 LC_NUMERIC=C
LC_TIME=English_United Kingdom.1252
attached base packages:
stats     graphics  grDevices utils     datasets  methods   base
other attached packages:
corrplot 0.84
  Taiyun Wei and Viliam Simko (2017). R package "corrplot": Visualization of
  a Correlation Matrix (Version 0.84). Available from
  https://github.com/taiyun/corrplot
MKmisc 1.6
  Kohl M (2019). MKmisc: Miscellaneous functions from M. Kohl_. R package
         version 1.6, http://www.stamats.de
epiR 1.0-14
  Mark Stevenson with contributions from Telmo Nunes, Cord Heuer, Jonathon
  Marshall, Javier Sanchez, Ron Thornton, Jeno Reiczigel, Jim Robison-Cox,
  Paola Sebastiani, Peter Solymos, Kazuki Yoshida, Geoff Jones, Sarah
  Pirikahu, Simon Firestone, Ryan Kyle, Johann Popp, Mathew Jay and Charles
  Reynard. (2020). epiR: Tools for the Analysis of Epidemiological Data. R
  package version 1.0-14. https://CRAN.R-project.org/package=epiR
Matching 4.9-7
  Jasjeet S. Sekhon (2011). Multivariate and Propensity Score Matching
  Software with Automated Balance Optimization: The Matching Package for R.
  Journal of Statistical Software, 42(7), 1-52. URL
         http://www.jstatsoft.org/v42/i07/.
MASS 7.3-51.5
  Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S.
  Fourth Edition. Springer, New York. ISBN 0-387-95457-0
Ordinal 2019.12-10
  Christensen, R. H. B. (2019). ordinal - Regression Models for Ordinal Data. R
         package version    2019.12-10. https://CRAN.R-
         project.org/package=ordinal.
Hmisc 4.4-0
  Frank E Harrell Jr, with contributions from Charles Dupont and many
  others. (2020). Hmisc: Harrell Miscellaneous. R package version 4.4-0.
  https://CRAN.R-project.org/package=Hmisc
Formula 1.2-3
  Achim Zeileis, Yves Croissant (2010). Extended Model Formulas in R:
  Multiple Parts and Multiple Responses. Journal of Statistical Software
  34(1), 1-13. doi:10.18637/jss.v034.i01
lattice 0.20-41
  Sarkar, Deepayan (2008) Lattice: Multivariate Data Visualization with R.
  Springer, New York. ISBN 978-0-387-75968-5
```

```
mice 3.8.0
  Stef van Buuren, Karin Groothuis-Oudshoorn (2011). mice: Multivariate
  Imputation by Chained Equations in R. Journal of Statistical Software,
  45(3), 1-67. URL https://www.jstatsoft.org/v45/i03/.
readxl 1.3.1
  Hadley Wickham and Jennifer Bryan (2019). readxl: Read Excel Files. R
  package version 1.3.1. https://CRAN.R-project.org/package=readxl
finalfit 1.0.1
  Ewen Harrison, Tom Drake and Riinu Ots (2020). finalfit: Quickly Create
  Elegant Regression Results Tables and Plots when Modelling. R package
  version 1.0.1. https://CRAN.R-project.org/package=finalfit
MatchIt 3.0.2
  Daniel E. Ho, Kosuke Imai, Gary King, Elizabeth A. Stuart (2011). MatchIt:
  Nonparametric Preprocessing for Parametric Causal Inference. Journal of
  Statistical Software, Vol. 42, No. 8, pp. 1-28. URL
  http://www.jstatsoft.org/v42/i08/
tableone 0.11.1
  Kazuki Yoshida (2020). tableone: Create 'Table 1' to Describe Baseline
  Characteristics. R package version 0.11.1.
  https://CRAN.R-project.org/package=tableone
forcats 0.5.0
  Hadley Wickham (2020). forcats: Tools for Working with Categorical
  Variables (Factors). R package version 0.5.0.
  https://CRAN.R-project.org/package=forcats
stringr 1.4.0
  Hadley Wickham (2019). stringr: Simple, Consistent Wrappers for Common
  String Operations. R package version 1.4.0.
  https://CRAN.R-project.org/package=stringr
dplyr 0.8.5
  Hadley Wickham, Romain François, Lionel Henry and Kirill Müller (2020).
  dplyr: A Grammar of Data Manipulation. R package version 0.8.5.
  https://CRAN.R-project.org/package=dplyr
purrr 0.3.4
  Lionel Henry and Hadley Wickham (2020). purrr: Functional Programming
  Tools. R package version 0.3.4. https://CRAN.R-project.org/package=purrr
readr 1.3.1
  Hadley Wickham, Jim Hester and Romain Francois (2018). readr: Read
  Rectangular Text Data. R package version 1.3.1.
  https://CRAN.R-project.org/package=readr
tidyr 1.0.2
  Hadley Wickham and Lionel Henry (2020). tidyr: Tidy Messy Data. R package
  version 1.0.2. https://CRAN.R-project.org/package=tidyr
tibble 3.0.0
  Hadley Wickham and Lionel Henry (2020). tidyr: Tidy Messy Data. R package
  version 1.0.2. https://CRAN.R-project.org/package=tidyr
ggplot2 3.3.0
  H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag
  New York, 2016.
tidyverse 1.3.0
  Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source
  Software, 4(43), 1686, https://doi.org/10.21105/joss.01686
forestplot 1.9
  Max Gordon and Thomas Lumley (2019). forestplot: Advanced Forest Plot Using
         'grid' Graphics. R package version 1.9.   https://CRAN.R-
         project.org/package=forestplot
MatchThem 0.9.3
  Farhad Pishgar and Noah Greifer (2020). MatchThem: Matching and Weighting
         Multiply Imputed Datasets. R package version 0.9.3. https://CRAN.R-
         project.org/package=MatchThem
```

```
miceadds 3.9-14
 Robitzsch, A., & Grund, S. (2020). miceadds: Some Additional Multiple
         Imputation Functions, Especially for 'mice'. R package version 3.9-14.
         https://CRAN.R-project.org/package=miceadds
cobalt 4.2.2
 Noah Greifer (2020). cobalt: Covariate Balance Tables and Plots. R package
         version 4.2.2.   https://CRAN.R-project.org/package=cobalt
```

1.1 Load Packages and Data

1.1.1 Load Packages:

```
library(MKmisc)
library(tidyverse)
library(tableone)
library(MatchIt)
library(finalfit)
library(readxl)
library(cobalt)
library(mice)
library(miceadds)
library(Hmisc)
library(epiR)
library(MatchThem)
library(ordinal)
library(forestplot)
```

1.2 Power Calculation

1.2.0.0.0.1 This code calculates the sample size (positive and negative by gold standard test) needed to evaluate a diagnostic test with 56% sensitivity at 80% power with alpha 0.05. The 56% value is the lower confidence reported by Wong et al. and lower sensitivities typically require higher sample sizes, the result is the same whether specificity or sensitivities are passed as arguments, the previously published specificities are higher than sensitivities so for a generous estimate, the sensitivity was used.

```
power <- power.diagnostic.test(sens = 0.56,
    sig.level = 0.05, delta = 0.1, power = 0.8) %>%
    print()
```

```
 Diagnostic test exact power calculation

           sens = 0.56
              n = 165
             n1 = 165
          delta = 0.1
      sig.level = 0.05
          power = 0.8
           prev = NULL

NOTE: n is number of cases, n1 is number of controls
```

2 Load Data:

```
data <- read_csv("FullDataWithCT.csv", col_types = cols(Age = col_integer(),
    Albumin = col_number(), CK = col_number(),
    CT = col_character(), CRP = col_number(),
    DDimer = col_number(), DateOfDeath = col_date(format = "%d/%m/%Y"),
    DateOfDischarge = col_date(format = "%d/%m/%Y"),
    DateOfVisit = col_date(format = "%d/%m/%Y"),
    DateOfSymptomOnset = col_date(format = "%d/%m/%Y"),
    DiastolicBP = col_number(), FiO2 = col_skip(),
    GCS = col_number(), HR = col_number(),
    MRN = col_skip(), NEWS = col_number(),
    `NEWS2(noFiO2)` = col_skip(), Neutrophils = col_number(),
    RR = col_number(), Sats = col_number(),
    `Supplemental Oxygen` = col_skip(), SystolicBP = col_number(),
    Temperature = col_number(), Troponin = col_number(),
    CTBSTI = col_integer()))
```

3 Data Cleaning

3.0.0.0.0.1 Format data into factors/ differences between dates:

```
data <- mutate_if(data, is.character, as.factor)
data$DayOfSymptoms <- difftime(data$DateOfVisit,
    data$DateOfSymptomOnset, units = "days")
data$TimeToDeath <- abs(difftime(data$DateOfDeath,
    data$DateOfVisit, units = "days"))
data$DayOfSymptoms <- as.numeric(data$DayOfSymptoms)
data$TimeToDeath <- as.numeric(data$TimeToDeath)
```

3.0.0.1 Recode ethnicities as too many options:

3.0.0.1.0.1 This code collapses the ethnicity categories into 'White', 'Black', 'South Asian', 'Other Asian', 'Mixed' or 'Other';

```
data$Ethnicity <- fct_collapse(data$Ethnicity,
    White = c("White - British", "White - Irish",
        "White - Any Other White Background"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    Black = c("Black - Any Other Black Background",
        "Black or Black British - A0rican",
        "Black or Black British - African",
        "Black or Black British - Caribbean"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    `South Asian` = c("Asian or Asian British - Bangladeshi",
        "Asian or Asian British - Indian",
        "Asian or Asian British - Pakistani"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    `Other Asian` = c("Asian - Any Other Asian Background",
        "Other - Chinese"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    Mixed = c("mixed - Any Other mixed Background",
        "Mixed - Any Other Mixed Background",
        "Mixed - White and Asian", "Mixed - White and Black African",
        "mixed - White and Black Caribbean",
        "Mixed - White and Black Caribbean"))
```

3.0.0.1.0.2 New XR positive column for "Classic Covid" or not:

```
data$XRPositive <- ifelse(data$XRChest ==
    "Classic COVID", "Positive", "Negative")
data$XRPositive <- as.factor(data$XRPositive)
```

3.0.1 Follow Up Swabs + Initial Swabs Positive:

3.0.1.0.0.1 Creates new column 'OverallPos' which includes initial RT-PCR swab and follow-up swabs in 30 days of attendance, if any are positive the value will be positive in this column

```
data$OverallPos <- case_when(data$RTPCR ==
    "Positive" | data$FollowUpPos == "Positive" ~
    "Positive")
data$OverallPos <- replace_na(data$OverallPos,
    "Negative")
```

3.0.1.0.0.2 Create new vector with all variable names (i.e. the column headers)

```
explanatory <- names(data)
```

3.0.2 Paired XR and RT-PCR data

3.0.2.1 Creates new variable 'completedata' which contains only patients who had both CXR and RT-PCR in ED

```
completedata <- filter(data, !is.na(data$XRPositive) &
    !is.na(data$RTPCR))
```

3.0.2.1.1 Remove missing data variable

```
completedata <- completedata[-c(31)]
```

3.0.2.2 Format complete data variables

```
completedata$OverallPos <- as.factor(completedata$OverallPos)

completedata$ThirtyDayFU <- as.factor(completedata$ThirtyDayFU)
completedata$TimeToDeath <- abs(difftime(completedata$DateOfDeath,
    completedata$DateOfVisit, units = "days"))

completedata$TimeToDeath <- as.numeric(completedata$TimeToDeath)
```

3.0.2.2.0.1 Set 'XRChest' as ordinal variable on scale of 'Alternative pathology' as lowest value and 'Classical COVID' as highest

```
completedata$XRChest <- ordered(completedata$XRChest,
    levels = c("Alternative pathology", "No abnormalities",
        "Indeterminate", "Classic COVID"))
```

3.0.2.2.0.2 Convert CT BSTI grade column into factor:

```
completedata$CTBSTI <- as.factor(completedata$CTBSTI)
```

4 Demographic table of raw data

4.0.0.0.0.1 This code creates an unformatted demographic table (table 2 in manuscript), for the raw data, stratified by RT-PCR status, significance testing between RT-PCR +ve and -ve groups is carried out automatically using chi squared, t-tests, ANOVA etc.; there is also a column for the proportion of missing data

```
CreateTableOne(vars = explanatory,
               strata = 'OverallPos',
               data = completedata) -> demogtable

#### List nonnormal factors for summarisation as median / IQR and non
        parametric statistical test

explanatorynnormal<-c("Sats","RR", "GCS", "SystolicBP", "Temperature", "HR",
        "Neutrophils",
+                      "DDimer","Albumin","CRP","CK","Troponin")
as.data.frame(print(demogtable, nonnormal = explanatorynnormal, missing =
        TRUE))->demogtable

write.csv(demogtable, file = "Demogtable.csv")

  Age (mean (SD))                    62.74 (17.72)             66.18 (17.58)
        0.001
  Ethnicity (%)
        0.097
     Other Asian                        29 ( 8.0)                 72 ( 11.8)
     South Asian                        27 ( 7.5)                 38 (  6.2)
     Black                              41 (11.4)                 91 ( 14.9)
     Mixed                               6 ( 1.7)                  6 (  1.0)
     Other - Any Other Ethnic Group     56 (15.5)                105 ( 17.2)
     White                             202 (56.0)                297 ( 48.8)
  Sex = Male (%)                       233 (53.6)                480 ( 62.9)
        0.002
  Sats (median [IQR])                95.00 [92.00, 98.00]      93.00 [88.00,
        96.00]    <0.001 nonnorm
  RR (median [IQR])                  22.00 [20.00, 28.00]      26.00 [20.00,
        32.00]    <0.001 nonnorm
  GCS (median [IQR])                 15.00 [15.00, 15.00]      15.00 [15.00,
        15.00]     0.043 nonnorm
  SystolicBP (median [IQR])         134.00 [119.00, 151.50]   130.00 [115.00,
        145.00]   0.009 nonnorm
  DiastolicBP (mean (SD))            79.54 (16.40)             75.61 (14.51)
        <0.001
  HR (median [IQR])                  96.00 [83.00, 110.00]     94.00 [81.00,
        108.00]    0.092 nonnorm
```

18 4 Demographic table of raw data

Temperature (median [IQR])	37.10 [36.60, 38.00]	37.70 [37.00, 38.40]	<0.001 nonnorm
XRChest (%)			<0.001
Alternative pathology	4 (0.9)	3 (0.4)	
No abnormalities	178 (40.9)	136 (17.8)	
Indeterminate	83 (19.1)	169 (22.1)	
Classic COVID	170 (39.1)	455 (59.6)	
CTPA = **PE** (%)	16 (30.2)	28 (45.9)	0.127
Comorbidity = **Yes** (%)	297 (79.0)	482 (80.3)	0.669
Dyspnoea = **Yes** (%)	274 (69.4)	497 (75.5)	0.034
Neutrophils (median [IQR])	6.42 [4.55, 9.11]	5.25 [3.69, 7.61]	<0.001 nonnorm
DDimer (median [IQR])	1250.00 [619.00, 3059.00]	1105.00 [626.00, 2428.50]	0.204 nonnorm
Albumin (median [IQR])	39.00 [35.00, 42.00]	37.00 [34.00, 40.00]	<0.001 nonnorm
CRP (median [IQR])	51.00 [13.00, 117.00]	83.00 [42.00, 158.00]	<0.001 nonnorm
CK (median [IQR])	91.00 [54.00, 169.00]	146.50 [78.00, 342.75]	<0.001 nonnorm
Troponin (median [IQR])	19.00 [7.00, 53.00]	20.00 [9.00, 53.00]	0.278 nonnorm
Admitted = **Discharged** (%)	104 (24.0)	128 (16.8)	0.003
AdmittedToITU = **Yes** (%)	5 (1.3)	32 (4.8)	0.005
RTPCR = **Positive** (%)	0 (0.0)	738 (96.7)	<0.001
CT = 1 (%)	37 (57.8)	26 (86.7)	0.011
NEWS (**mean** (SD))	4.36 (3.06)	5.48 (2.71)	0.032
ThirtyDayFU (%)			<0.001
1	219 (78.2)	367 (58.3)	
2	14 (5.0)	49 (7.8)	
3	18 (6.4)	60 (9.5)	
4	29 (10.4)	154 (24.4)	
CTBSTI (%)			<0.001
0	23 (22.1)	6 (3.3)	
1	52 (50.0)	157 (85.8)	
2	14 (13.5)	14 (7.7)	
3	15 (14.4)	6 (3.3)	
DayOfSymptoms (**mean** (SD))	9.84 (9.63)	8.56 (15.80)	0.368
TimeToDeath (**mean** (SD))	50.33 (77.93)	57.76 (70.02)	0.618
XRPositive = **Positive** (%)	170 (39.1)	455 (59.6)	<0.001
OverallPos = **Positive** (%)	0 (0.0)	763 (100.0)	

`

4.0.0.0.0.2 Limited dataset comprising relevant data and those without significant missingness:

```
limcompletedata <- dplyr::select(completedata,
    c("Age", "XRChest", "Ethnicity", "Sex",
        "RR", "Sats", "GCS", "Temperature",
        "HR", "SystolicBP", "DiastolicBP",
        "Neutrophils", "DDimer", "CRP", "Troponin",
        "Albumin", "CK", "OverallPos", "Admitted",
        "AdmittedToITU", "ThirtyDayFU", "Dyspnoea",
        "Comorbidity", "XRPositive"))
```

5 Imputation

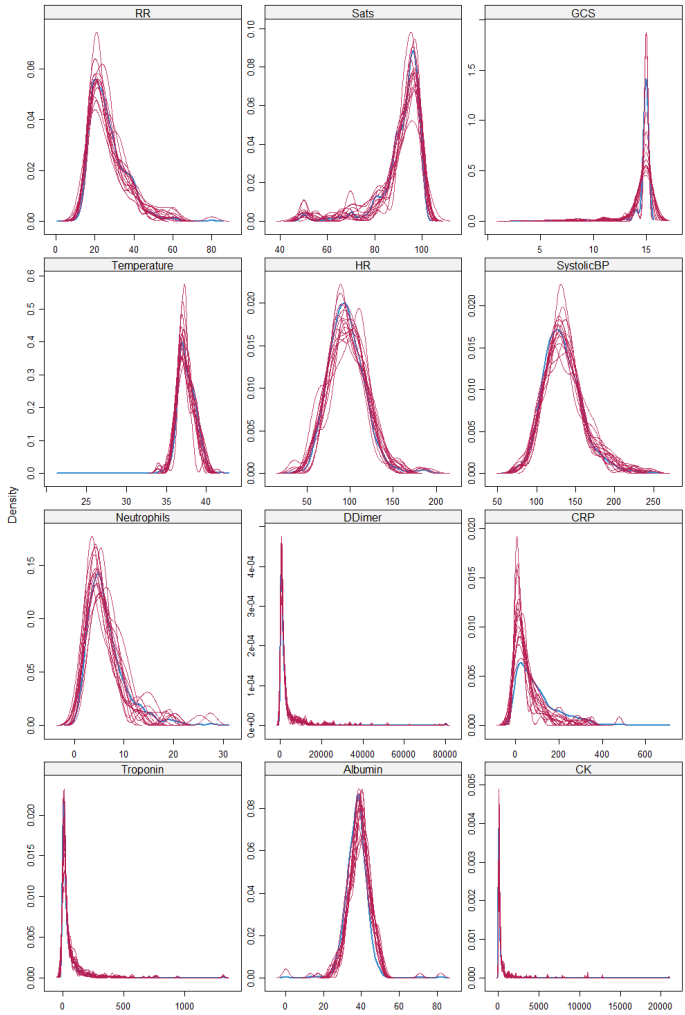
5.0.0.0.0.1 This code generates 15 imputed datasets using the permuted mean matching method, based on the 'limcompletedata' dataset which has filtered the most relevant fields, with minimal missing data initially

```
imputed <- mice(limcompletedata, m = 15,
    method = "pmm")
```

5.0.0.0.0.2 Imputation Diagnostics Density plot, this corresponds to supplementary figure 1:

```
densityplot(imputed)
```

22 5 Imputation



6 Propensity Score Matching

6.0.0.0.0.1 This code matches data in the imputed datasets on whether the XR was reported classical COVID or not, the matching is done based on the covariates Sex, Age, Comorbidity, Ethnicity and Respiratory Rate

```
library(MatchThem)
#### MatchThem package requires dependent variable to be coded as 0 or 1
imputed[["data"]][["XRPositive"]] %>% recode_factor("Positive" = "1",
         "Negative" = "0") ->imputed[["data"]][["XRPositive"]]
matchthem(
  XRPositive ~ Sex + Age + Comorbidity + Ethnicity + RR,
  data = imputed,
  method = 'nearest',
  verbose = FALSE,
  replace = FALSE,
  ratio = 1,
  caliper = 0.2,
  m.order = "random",) -> matchedtest
### Set XRChest to unordered for binomial analyses
matchedtest[["datasets"]]c(1:15)[["XRChest"]] %>% factor(ordered = FALSE) ->
         matched2[["datasets"]]c(1:15)[["XRChest"]]
```

6.1 Match Balance Diagnostics

6.1.0.0.1 Creates plots and table with mean difference and distributation of values in covariates betweeen XR +ve and -ve groups after matching across all imputed datasets:

```
#### Supplementary tables 1,2 and 3:
bal.tab(matchedtest)
#### Supplementary figure 2
bal.plot(matchedtest)
#### Supplementary figure 3:
bal.plot(matchedtest, var.name = "Age", type = "histogram",
    which = "both")
bal.plot(matchedtest, var.name = "Sex", type = "histogram",
    which = "both")
bal.plot(matchedtest, var.name = "Ethnicity",
```

```
    type = "histogram", which = "both")
bal.plot(matchedtest, var.name = "RR", type = "histogram",
    which = "both")
bal.plot(matchedtest, var.name = "Comorbidity",
    type = "histogram", which = "both")
#### Supplementary figure 4:
love.plot(matchedtest)
```

7 Matched Demographics Table:

7.0.0.0.0.1 Stack matched imputed datasets into one large datset and split into COVID +ve and -ve groups:

```
### 'all=FALSE' gets matched data only
stacked <- MatchThem::complete(matchedtest,
    n = c(1:15), all = FALSE)
stacked <- stacked %>% filter(.imp > 0)
```

7.0.0.0.0.2 Creates demographics table as above, but on propensity matched imputed datasets, corresponds to Table 4:

```
table4 <- CreateTableOne(strata = "OverallPos",
    data = stacked)
#### Means and SD kept as is, mean counts
#### calculated after dividing by 15 (as 15
#### imputed datasets)
```

7.0.0.0.0.3 Creates demographic table stratified by XR Positive or Negative on matched imputed datasets, correpsonds to Table 5:

```
table5 <- CreateTableOne(strata = "XRPositive",
    data = stacked)
#### Means and SD kept as is, mean counts
#### calculated after dividing by 15 (as 15
#### imputed datasets)
```

7.0.0.0.0.4 Summary statistics for pooled data:

```
### Normal means sd
explanatorynorm <- c("Age", "Temperature",
    "HR", "SystolicBP")
summarynormalOverallPos <- stacked %>% group_by(OverallPos) %>%
```

26 7 Matched Demographics Table:

```
    summarise_at(vars(explanatorynorm), list(mean.default,
        sd))
summarynormalXRPositive <- stacked %>% group_by(XRPositive) %>%
    summarise_at(vars(explanatorynorm), list(mean.default,
        sd))

### Non normal medians and IQR
summarynnormalOverallPos <- stacked %>% group_by(OverallPos) %>%
    summarise_at(vars(explanatorynnormal),
        list(median, IQR))
summarynnormalXRPositive <- stacked %>% group_by(XRPositive) %>%
    summarise_at(vars(explanatorynnormal),
        list(median, IQR))
```

8 Diagnostic Accuracy

8.0.0.1 This section generates the diagnostic accuracy statistics (e.g. sensitivity, specificity) for CXR and CT with RT-PCR as the reference standard using the matched imputed datasets

8.0.0.2 This code creates a contingency table of False/ True Positives and Negatives for Chest X-ray taken from the demographic tables above:

```
contingxr <- matrix(c(305, 243, 125, 187),
    nrow = 2, ncol = 2)

colnames(contingxr) <- c("PCR+", "PCR-")

rownames(contingxr) <- c("XR+", "XR-")
```

8.0.0.2.1 This function calculates diagnostic accuracy test statistics:

```
xraccuracy <- epi.tests(contingxr, conf.level = 0.95)
```

8.0.0.3 Giving the diagnostic accuracy output for CXR in table 3:

```
xraccuracy
          Outcome +    Outcome -      Total
Test +          305          125        430
Test -          243          187        430
Total           548          312        860

Point estimates and 95 % CIs:
-------------------------------------------------------------
Apparent prevalence                   0.50 (0.47, 0.53)
True prevalence                       0.64 (0.60, 0.67)
```

```
Sensitivity                          0.56 (0.51, 0.60)
Specificity                          0.60 (0.54, 0.65)
Positive predictive value            0.71 (0.66, 0.75)
Negative predictive value            0.43 (0.39, 0.48)
Positive likelihood ratio            1.39 (1.19, 1.62)
Negative likelihood ratio            0.74 (0.65, 0.84)
-------------------------------------------------------```
```

8.0.0.3.0.1 NB diagnostic accuracy values in table available in list view of xraccuracy variable

---

8.1 CT Data and Accuracy

8.1.0.0.0.1 Only those with CT and RT PCR:

```
CTdata <- filter(data, is.na(data$CTBSTI) ==
    FALSE & is.na(data$RTPCR) == FALSE)
```

8.1.0.0.0.2 Select relevant variables

```
CTdata <- dplyr::select(CTdata, c("Age",
    "XRChest", "Ethnicity", "Sex", "RR",
    "Sats", "GCS", "Temperature", "HR", "SystolicBP",
    "DiastolicBP", "Neutrophils", "DDimer",
    "CRP", "Troponin", "OverallPos", "Admitted",
    "AdmittedToITU", "ThirtyDayFU", "Dyspnoea",
    "Comorbidity", "XRPositive", "OverallPos",
    "CTBSTI"))
```

8.1.0.0.0.3 Set RT-PCR as factor:

```
CTdata$OverallPos <- as.factor(CTdata$OverallPos)
```

8.1.0.0.0.4 Rename 1 and 0 to Positive and Negative:

```
CTdata$CTPositive <- ifelse(CTdata$CTBSTI ==
    "1", "Positive", "Negative")
CTdata$CTPositive <- as.factor(CTdata$CTPositive)
```

8.1.0.0.0.5 Regression with CT as outcome variable:

```
CT <- finalfit(
  CTdata,
  "OverallPos",
  c(
    "Age",
    "Sex",
    "RR",
    "GCS",
    "CTPositive",
    "Temperature",
    "HR",
    "SystolicBP",
    "DiastolicBP",
    "Sats",
    "Dyspnoea",
    "Comorbidity"

  ),
  confint_level = 0.95
)
```

8.1.0.0.0.6 Contingency table of True/False Positives and Negatives for CT taken from Regression table:

```
contingct <- matrix(c(CT[7, 4], CT[6, 4],
    CT[7, 3], CT[6, 3]), nrow = 2, ncol = 2)
colnames(contingct) <- c("PCR+", "PCR-")
rownames(contingct) <- c("CT+", "CT-")
contingct <- substr(contingct, start = 1,
    stop = 3)
contingct <- sapply(contingct, as.numeric)
contingct <- matrix(contingct, nrow = 2,
    ncol = 2)
colnames(contingct) <- c("PCR+", "PCR-")

rownames(contingct) <- c("CT+", "CT-")
```

8.1.0.0.0.7 Diagnostic accuracy statistics for CT

```
epi.tests(contingct, conf.level = 0.95) -> ctaccuracy
          Outcome +    Outcome -      Total
Test +          162           55        217
Test -           29           56         85
Total           191          111        302

Point estimates and 95 % CIs:
-------------------------------------------------------------
Apparent prevalence                    0.72 (0.66, 0.77)
True prevalence                        0.63 (0.58, 0.69)
Sensitivity                            0.85 (0.79, 0.90)
Specificity                            0.50 (0.41, 0.60)
Positive predictive value              0.75 (0.68, 0.80)
Negative predictive value              0.66 (0.55, 0.76)
Positive likelihood ratio              1.71 (1.41, 2.08)
Negative likelihood ratio              0.30 (0.21, 0.44)
-------------------------------------------------------------
```

8.1.0.0.0.8 NB Diagnostic accuracy values found in list view rather than output

---

8.2 CT and XR accuracy comparison

8.2.0.1 In this section mean differences of diagnostic accuracy statistics between CT and Chest X-ray with confidence intervals and p-values are calculated

8.2.1 Sensitivity

8.2.1.0.0.1 Upper confidence limit for difference in sensitivity

```
ubsens <- (ctaccuracy[["elements"]][["se.up"]] -
    xraccuracy[["elements"]][["se.low"]])
```

8.2.1.0.0.2 Lower confidence limit for difference in sensitivity

```
lbsens <- (ctaccuracy[["elements"]][["se.low"]] -
    xraccuracy[["elements"]][["se.up"]])
```

8.2.1.0.0.3 Mean difference in sensitivity

```
meansens <- ctaccuracy[["elements"]][["se"]] -
    xraccuracy[["elements"]][["se"]]
```

8.2.1.0.0.4 Standard error for sensitivity

```
sesens <- (ubsens - lbsens)/(2 * 1.96)
```

8.2.1.0.0.5 value for difference in sensitivity

```
zsens <- meansens/sesens
```

8.2.1.0.0.6 P-value for difference in sensitivity

```
psens <- exp(-0.717 * zsens - 0.416 * zsens^2)
```

8.2.1.0.0.7 Format values into 'mean difference (95% CI) p-value' rounded to 2 d.p.

```
diffsens <- sprintf("%s (%s-%s)", round(meansens,
    digits = 2), round(lbsens, digits = 2),
    round(ubsens, digits = 2))
diffsensp <- c(diffsens, psens)
```

8.2.1.0.0.8 Subsequent analyses in this section follow the code above

```
## Specificity
ubspec <- (ctaccuracy[["elements"]][["sp.up"]] -
    xraccuracy[["elements"]][["sp.low"]])
lbspec <- (ctaccuracy[["elements"]][["sp.low"]] -
    xraccuracy[["elements"]][["sp.up"]])
meanspec <- ctaccuracy[["elements"]][["sp"]] -
    xraccuracy[["elements"]][["sp"]]
sespec <- (ubspec - lbspec)/(2 * 1.96)
zspec <- meanspec/sespec
pspec <- exp(-0.717 * zspec - 0.416 * zspec^2)
diffspec <- sprintf("%s (%s-%s)", round(meanspec,
    digits = 2), round(lbspec, digits = 2),
    round(ubspec, digits = 2))
diffspecp <- c(diffspec, pspec)


ubda <- (ctaccuracy[["elements"]][["da.up"]] -
    xraccuracy[["elements"]][["da.low"]])
lbda <- (ctaccuracy[["elements"]][["da.low"]] -
    xraccuracy[["elements"]][["da.up"]])
meanda <- ctaccuracy[["elements"]][["da"]] -
    xraccuracy[["elements"]][["da"]]
seda <- (ubda - lbda)/(2 * 1.96)
zda <- meanda/seda
pda <- exp(-0.717 * zda - 0.416 * zda^2)
diffda <- sprintf("%s (%s-%s)", round(meanda,
    digits = 2), round(lbda, digits = 2),
    round(ubda, digits = 2))
diffdap <- c(diffda, pda)
## Positive Likelihood Ratio
ublrpos <- (ctaccuracy[["elements"]][["lrpos.up"]] -
    xraccuracy[["elements"]][["lrpos.low"]])
lblrpos <- (ctaccuracy[["elements"]][["lrpos.low"]] -
    xraccuracy[["elements"]][["lrpos.up"]])
meanlrpos <- ctaccuracy[["elements"]][["lrpos"]] -
    xraccuracy[["elements"]][["lrpos"]]
selrpos <- (ublrpos - lblrpos)/(2 * 1.96)
zlrpos <- meanlrpos/selrpos
plrpos <- exp(-0.717 * zlrpos - 0.416 * zlrpos^2)
difflrpos <- sprintf("%s (%s-%s)", round(meanlrpos,
    digits = 2), round(lblrpos, digits = 2),
```

```
    round(ublrpos, digits = 2))
difflrposp <- c(difflrpos, plrpos)
## Negative Likelihood Ratios
ublrneg <- (ctaccuracy[["elements"]][["lrneg.up"]] -
    xraccuracy[["elements"]][["lrneg.low"]])
lblrneg <- (ctaccuracy[["elements"]][["lrneg.low"]] -
    xraccuracy[["elements"]][["lrneg.up"]])
meanlrneg <- ctaccuracy[["elements"]][["lrneg"]] -
    xraccuracy[["elements"]][["lrneg"]]
selrneg <- (ublrneg - lblrneg)/(2 * 1.96)
zlrneg <- meanlrneg/selrneg
plrneg <- exp(-0.717 * zlrneg - 0.416 * zlrneg^2)
difflrneg <- sprintf("%s (%s-%s)", round(meanlrneg,
    digits = 2), round(lblrneg, digits = 2),
    round(ublrneg, digits = 2))
difflrnegp <- c(difflrneg, plrneg)


## Positive Predictive Value
ppv <- (ctaccuracy[["elements"]][["ppv.low"]] -
    xraccuracy[["elements"]][["ppv.up"]])
meanppv <- ctaccuracy[["elements"]][["ppv"]] -
    xraccuracy[["elements"]][["ppv"]]
seppv <- (ubppv - lbppv)/(2 * 1.96)
zppv <- meanppv/seppv
pppv <- exp(-0.717 * zppv - 0.416 * zppv^2)
diffppv <- sprintf("%s (%s-%s)", round(meanppv,
    digits = 2), round(lbppv, digits = 2),
    round(ubppv, digits = 2))
diffppvp <- c(diffppv, pppv)

## Negative Predictive Value
npv <- (ctaccuracy[["elements"]][["npv.low"]] -
    xraccuracy[["elements"]][["npv.up"]])
meannpv <- ctaccuracy[["elements"]][["npv"]] -
    xraccuracy[["elements"]][["npv"]]
senpv <- (ubnpv - lbnpv)/(2 * 1.96)
znpv <- meannpv/senpv
pnpv <- exp(-0.717 * znpv - 0.416 * znpv^2)
diffnpv <- sprintf("%s (%s-%s)", round(meannpv,
    digits = 2), round(lbnpv, digits = 2),
    round(ubnpv, digits = 2))
diffnpvp <- c(diffnpv, pnpv)


## Apparent Prevalence
meantp <- ctaccuracy[["elements"]][["tp"]] -
    xraccuracy[["elements"]][["tp"]]
setp <- (ubtp - lbtp)/(2 * 1.96)
ztp <- meantp/setp
ptp <- exp(-0.717 * ztp - 0.416 * ztp^2)
difftp <- sprintf("%s (%s-%s)", round(meantp,
    digits = 2), round(lbtp, digits = 2),
    round(ubtp, digits = 2))
difftpp <- c(difftp, ptp)

## True Prevalence
meanap <- ctaccuracy[["elements"]][["ap"]] -
    xraccuracy[["elements"]][["ap"]]
```

```
seap <- (ubap - lbap)/(2 * 1.96)
zap <- meanap/seap
pap <- exp(-0.717 * zap - 0.416 * zap^2)
diffap <- sprintf("%s (%s-%s)", round(meanap,
    digits = 2), round(lbap, digits = 2),
    round(ubap, digits = 2))
diffapp <- c(diffap, pap)
```

8.3 Intermodality Agreement

8.3.0.0.0.1 This section contains code to analyse the level of agreement in the unmatched CT dataset which contains only data with CT, XR and RT-PCR

8.3.0.0.0.2 First- comparing CT and XR agreement

```
library(irr)
kappa2(c(CTdata$XRPositive, CTdata$CTPositive),
    weight = "squared")
d <- CTdata %>% select(c("CTPositive", "XRPositive"))
View(d)
kappa2(d, weight = "squared")
```

8.3.0.0.0.3 Output:

```
 Cohen's Kappa for 2 Raters (Weights: squared)

 Subjects = 287
   Raters = 2
    Kappa = 0.406

        z = 7.14
  p-value = 9.37e-13
```

8.3.0.0.0.4 The following code compares RT-PCR, CT and XR

```
d2 <- CTdata %>% select(c("CTPositive", "XRPositive",
    "OverallPos"))
View(d2)
kappam.fleiss(d2)
```

8.3.0.0.0.5 Output:

```
Fleiss' Kappa for m Raters

Subjects = 287
  Raters = 3
   Kappa = 0.361

       z = 10.6
 p-value = 0
```

8.3.1 Diagnostic Accuracy Analysis when Indeterminate Reports of CXR and CT are taken as positive

8.3.1.1 XR Indeterminates

8.3.1.1.0.1 New column for positive if indeterminate

```
stacked$XRIndPositive <- ifelse(stacked$XRChest ==
    "Classic COVID" | stacked$XRChest ==
    "Indeterminate", "Positive", "Negative")
stacked$XRIndPositive <- as.factor(stacked$XRIndPositive)
stackedpos <- stacked %>% filter(OverallPos ==
    "Positive")
stackedneg <- stacked %>% filter(OverallPos ==
    "Negative")
summary(stackedpos$XRIndPositive)
summary(stackedneg$XRIndPositive)

contingxrind <- matrix(c(441, 107, 186, 126),
    nrow = 2, ncol = 2)
colnames(contingxrind) <- c("PCR+", "PCR-")

rownames(contingxrind) <- c("XR+", "XR-")
xrindaccuracy <- epi.tests(contingxrind)
```

8.3.1.1.0.2 In this section mean differences of diagnostic accuracy statistics between CT (when CT indeterminates are not counted as positive)and Chest X-ray with confidence intervals and p-values are calculated, follows the same pattern as code previously

```
###### Sensitivity Upper confidence limit for
###### difference in sensitivity

ubsens <- (ctaccuracy[["elements"]][["se.up"]] -
    xrindaccuracy[["elements"]][["se.low"]])
## Lower confidence limit for difference
## in sensitivity
lbsens <- (ctaccuracy[["elements"]][["se.low"]] -
    xrindaccuracy[["elements"]][["se.up"]])
## Mean difference in sensitivity
meansens <- ctaccuracy[["elements"]][["se"]] -
    xrindaccuracy[["elements"]][["se"]]
## Standard error for sensitivity
sesens <- (ubsens - lbsens)/(2 * 1.96)
## Z value for difference in sensitivity
zsens <- meansens/sesens
## P-value for difference in sensitivity
psens <- exp(-0.717 * zsens - 0.416 * zsens^2)
### Format values into 'mean difference
### (95% CI) p-value' rounded to 2 d.p.
diffsens <- sprintf("%s (%s-%s)", round(meansens,
    digits = 2), round(lbsens, digits = 2),
    round(ubsens, digits = 2))
diffsensp <- c(diffsens, psens)

### Subsequent analyses in this section
### follow the code above Specificity
ubspec <- (ctaccuracy[["elements"]][["sp.up"]] -
    xrindaccuracy[["elements"]][["sp.low"]])
lbspec <- (ctaccuracy[["elements"]][["sp.low"]] -
    xrindaccuracy[["elements"]][["sp.up"]])
meanspec <- ctaccuracy[["elements"]][["sp"]] -
    xrindaccuracy[["elements"]][["sp"]]
sespec <- (ubspec - lbspec)/(2 * 1.96)
zspec <- meanspec/sespec
pspec <- exp(-0.717 * zspec - 0.416 * zspec^2)
diffspec <- sprintf("%s (%s-%s)", round(meanspec,
    digits = 2), round(lbspec, digits = 2),
    round(ubspec, digits = 2))
diffspecp <- c(diffspec, pspec)


ubda <- (ctaccuracy[["elements"]][["da.up"]] -
    xrindaccuracy[["elements"]][["da.low"]])
lbda <- (ctaccuracy[["elements"]][["da.low"]] -
    xrindaccuracy[["elements"]][["da.up"]])
meanda <- ctaccuracy[["elements"]][["da"]] -
    xrindaccuracy[["elements"]][["da"]]
seda <- (ubda - lbda)/(2 * 1.96)
```

```
zda <- meanda/seda
pda <- exp(-0.717 * zda - 0.416 * zda^2)
diffda <- sprintf("%s (%s-%s)", round(meanda,
    digits = 2), round(lbda, digits = 2),
    round(ubda, digits = 2))
diffdap <- c(diffda, pda)
## Positive Likelihood Ratio
ublrpos <- (ctaccuracy[["elements"]][["lrpos.up"]] -
    xrindaccuracy[["elements"]][["lrpos.low"]])
lblrpos <- (ctaccuracy[["elements"]][["lrpos.low"]] -
    xrindaccuracy[["elements"]][["lrpos.up"]])
meanlrpos <- ctaccuracy[["elements"]][["lrpos"]] -
    xrindaccuracy[["elements"]][["lrpos"]]
selrpos <- (ublrpos - lblrpos)/(2 * 1.96)
zlrpos <- meanlrpos/selrpos
plrpos <- exp(-0.717 * zlrpos - 0.416 * zlrpos^2)
difflrpos <- sprintf("%s (%s-%s)", round(meanlrpos,
    digits = 2), round(lblrpos, digits = 2),
    round(ublrpos, digits = 2))
difflrposp <- c(difflrpos, plrpos)
## Negative Likelihood Ratios
ublrneg <- (ctaccuracy[["elements"]][["lrneg.up"]] -
    xrindaccuracy[["elements"]][["lrneg.low"]])
lblrneg <- (ctaccuracy[["elements"]][["lrneg.low"]] -
    xrindaccuracy[["elements"]][["lrneg.up"]])
meanlrneg <- ctaccuracy[["elements"]][["lrneg"]] -
    xrindaccuracy[["elements"]][["lrneg"]]
selrneg <- (ublrneg - lblrneg)/(2 * 1.96)
zlrneg <- meanlrneg/selrneg
plrneg <- exp(-0.717 * zlrneg - 0.416 * zlrneg^2)
difflrneg <- sprintf("%s (%s-%s)", round(meanlrneg,
    digits = 2), round(lblrneg, digits = 2),
    round(ublrneg, digits = 2))
difflrnegp <- c(difflrneg, plrneg)

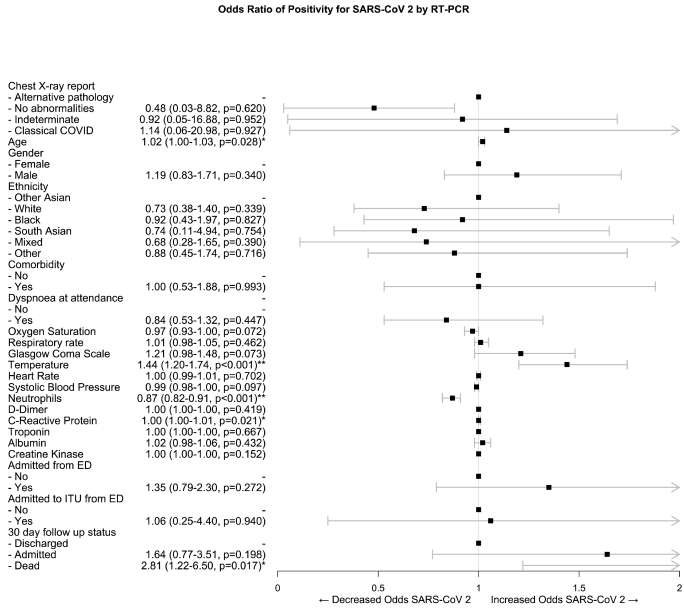

## Positive Predictive Value
ppv <- (ctaccuracy[["elements"]][["ppv.low"]] -
    xrindaccuracy[["elements"]][["ppv.up"]])
meanppv <- ctaccuracy[["elements"]][["ppv"]] -
    xrindaccuracy[["elements"]][["ppv"]]
seppv <- (ubppv - lbppv)/(2 * 1.96)
zppv <- meanppv/seppv
pppv <- exp(-0.717 * zppv - 0.416 * zppv^2)
diffppv <- sprintf("%s (%s-%s)", round(meanppv,
    digits = 2), round(lbppv, digits = 2),
    round(ubppv, digits = 2))
diffppvp <- c(diffppv, pppv)

## Negative Predictive Value
npv <- (ctaccuracy[["elements"]][["npv.low"]] -
    xrindaccuracy[["elements"]][["npv.up"]])
meannpv <- ctaccuracy[["elements"]][["npv"]] -
    xrindaccuracy[["elements"]][["npv"]]
senpv <- (ubnpv - lbnpv)/(2 * 1.96)
znpv <- meannpv/senpv
pnpv <- exp(-0.717 * znpv - 0.416 * znpv^2)
diffnpv <- sprintf("%s (%s-%s)", round(meannpv,
    digits = 2), round(lbnpv, digits = 2),
```

```
    round(ubnpv, digits = 2))
diffnpvp <- c(diffnpv, pnpv)

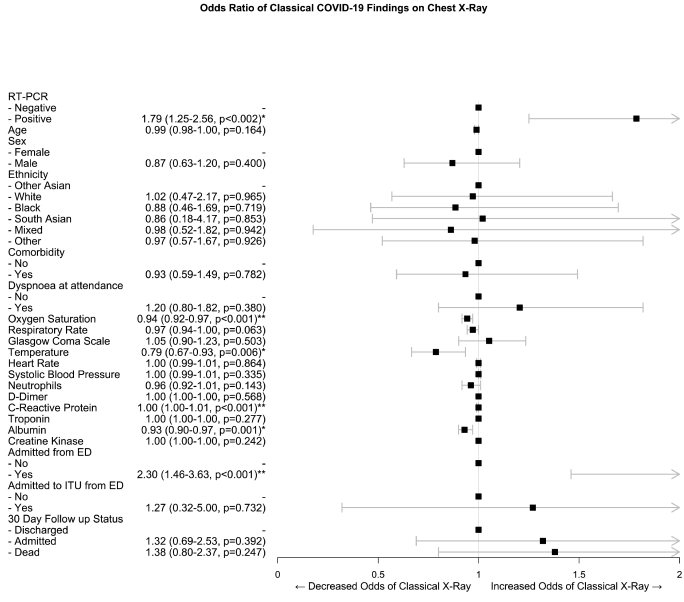

## True Prevalence
meantp <- ctaccuracy[["elements"]][["tp"]] -
    xrindaccuracy[["elements"]][["tp"]]
setp <- (ubtp - lbtp)/(2 * 1.96)
ztp <- meantp/setp
ptp <- exp(-0.717 * ztp - 0.416 * ztp^2)
difftp <- sprintf("%s (%s-%s)", round(meantp,
    digits = 2), round(lbtp, digits = 2),
    round(ubtp, digits = 2))
difftpp <- c(difftp, ptp)

## Apparent Prevalence
meanap <- ctaccuracy[["elements"]][["ap"]] -
    xrindaccuracy[["elements"]][["ap"]]
seap <- (ubap - lbap)/(2 * 1.96)
zap <- meanap/seap
pap <- exp(-0.717 * zap - 0.416 * zap^2)
diffap <- sprintf("%s (%s-%s)", round(meanap,
    digits = 2), round(lbap, digits = 2),
    round(ubap, digits = 2))
diffapp <- c(diffap, pap)
```

8.3.1.2 CT Indeterminates

8.3.1.2.0.1 New column for positive if indeterminate

```
CTdata$CTIndPositive <- ifelse(CTdata$CTBSTI ==
    "1" | CTdata$CTBSTI == "2", "Positive",
    "Negative")
CTdata$CTIndPositive <- as.factor(CTdata$CTIndPositive)
valuesctind <- CTdata %>% group_by(OverallPos,
    CTIndPositive) %>% summarise(n = n())
ctcontingind <- matrix(data = c(178, 13,
    70, 41), nrow = 2, ncol = 2)

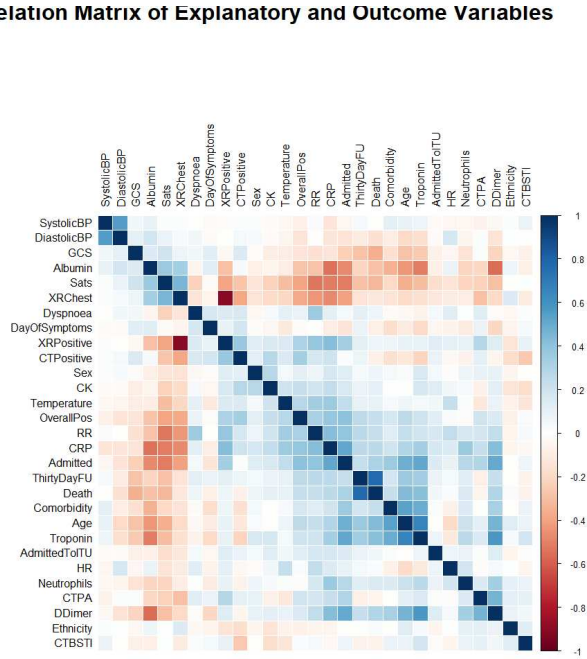
colnames(ctcontingind) <- c("PCR+ve", "PCR-ve")
rownames(ctcontingind) <- c("CT+ve", "CT-ve")
ctindaccuracy <- epi.tests(ctcontingind)
```

9 Pooled Regression after Multiple Imputation and Propensity Score Matching

9.0.0.0.0.1 Binomnal Logistic regression with RT-PCR as dependent variable

```
overallposmatchimp <- matchedtest %>% with(glm(formula(ff_formula(dependent =
         "OverallPos",
    explanatory = c("Age", "Ethnicity", "Sex",
        "RR", "GCS", "Temperature", "HR",
        "SystolicBP", "Neutrophils", "DDimer",
        "CRP", "Troponin", "Albumin", "CK",
        "Sats", "Admitted", "AdmittedToITU",
        "ThirtyDayFUTwo", "Dyspnoea", "Comorbidity",
        "XRChest"))), family = "binomial"),
    all = FALSE)
P <- overallposmatchimp %>% pool()
multivarpooledoverallpos = P %>% fit2df(estimate_name = "OR (multiple
         imputation)",
    exp = TRUE)
```

9.0.0.0.0.2 'multivarpooledoverallpos' produces multivariate odds ratios for each explanatory variable, corresponding to Table 4

9.0.1 Pooled Univariate Odds Ratios for OverallPos as dependent variable

9.0.1.0.0.1 This code is run with each of the explanatory variables in table 4 as arguments to produce their respective odds Ratios in table 4

```
overallposmatchimpunivar <- matchedtest %>%
    with(glm(formula(ff_formula(dependent = "OverallPos",
```

```
        explanatory = "XRChest")), family = "binomial"))
P <- overallposmatchimpunivar %>% pool()
univarpooledoverallpos = univaroverallpos <- P %>%
    fit2df(estimate_name = "OR (univariate)",
        exp = TRUE)
univaroverallpos
```

9.0.2 Binomial Logistic Regression with Positive Chest X-ray Report as Dependent Variable

9.0.2.0.0.1 This code follows the format above to produce univariate and multivariate odds ratios for each explanatory variable for having a positive XR report

9.0.3 Univariate XRPositive as dependent

9.0.3.0.0.1 (different explanatory variables passed into function to produce Odds ratios for each)

```
XRChestmatchimp <- matchedtest %>% with(glm(formula(ff_formula(dependent =
         "XRPositive",
    explanatory = "Comorbidity")), family = "binomial"))
P <- XRChestmatchimp %>% pool()
multivarpooledXRChest = univarXRChest <- P %>%
    fit2df(estimate_name = "OR (univariate)",
        exp = TRUE)
univarXRChest
```

9.0.4 Multivariate XRPositive as dependent

```
XRChestmatchimp <- matchedtest %>% with(glm(formula(ff_formula(dependent =
         "XRPositive",
    explanatory = c("Age", "OverallPos",
        "Ethnicity", "Sex", "RR", "GCS",
        "Temperature", "HR", "SystolicBP",
        "Neutrophils", "DDimer", "CRP", "Troponin",
        "Albumin", "CK", "Sats", "Admitted",
        "AdmittedToITU", "ThirtyDayFUTwo",
        "Dyspnoea", "Comorbidity"))), family = "binomial"))
P <- XRChestmatchimp %>% pool()
multivarpooledXRChest = multivarXRChest <- P %>%
    fit2df(estimate_name = "OR (multivariate)",
```

```
        exp = TRUE)
multivarXRChest
```

9.0.5 Pooled Ordinal Logistic Regression with XRPositive as dependent

9.0.5.0.0.1 This code also produces multivariate odds ratios for table 5, however, uses ordinal linear regression after the CXR report variable is converted to an ordered categorical variable, with alternative pathology as the lowest and classic covid as the highest value (see table 3)

```
XRChestmatchimpord <- matchedtest %>% with(clm(formula = XRChest ~
    Age + OverallPos + Ethnicity + Sex +
        RR + GCS + Temperature + HR + SystolicBP +
        Neutrophils + DDimer + CRP + Troponin +
        Sats + Admitted + AdmittedToITU +
        ThirtyDayFUTwo + Dyspnoea + Comorbidity))
P <- pool(object = XRChestmatchimpord[["analyses"]])
multivarpooledXRChestord = multivarXRChestord <- P %>%
    fit2df(estimate_name = "OR (multivariate)",
        exp = TRUE)
multivarXRChestord
```

9.1 Forest Plots

9.1.0.0.0.1 Creates forest plots for post matched regression tables above:

```
Figure1Forest <- read_excel("Figure1Forest.xlsx",
    col_types = c("text", "numeric", "numeric",
        "numeric", "text", "text"))

tabletext1 <- cbind(Figure1Forest$explanatory,
    Figure1Forest$summary)
forestplot(tabletext1, Figure1Forest$Mean,
    Figure1Forest$Lower, Figure1Forest$Upper,
    is.summary = FALSE, clip = c(0, 2), xlab = "<U+2190> Decreased Odds SARS-
        CoV 2       Increased Odds SARS-CoV 2 <U+2192>",
    zero = 1, cex = 0.9, lineheight = unit(6,
        "mm"), boxsize = 0.4, colgap = unit(6,
        "mm"), lwd.ci = 2, ci.vertices = TRUE,
    ci.vertices.height = 0.4, title = "Odds Ratio of Positivity for SARS-CoV 2
        by RT-PCR",
    txt_gp = fpTxtGp(label = gpar(cex = 1.25),
        ticks = gpar(cex = 1.1), xlab = gpar(cex = 1.2),
```

```
        title = gpar(cex = 1.2)), graphwidth = unit(200,
        "mm"))
```

9.1.0.0.0.2 Figure 2:



9.1.0.0.0.3 Figure 3 (XR dependent):

```
Figure2Forest <- read_excel("Figure2Forest.xlsx",
                            col_types = c("text", "numeric", "numeric",
                                          "numeric", "text", "text"))

tabletext2<-cbind(Figure2Forest$explanatory,Figure2Forest$summary)
forestplot (tabletext2, Figure2Forest$Mean,
            Figure2Forest$Lower, Figure2Forest$Upper, is.summary = FALSE,
            clip = c(0, 2),
```

9.1 Forest Plots 43

```
            xlab="\u2190 Decreased Odds of Classical X-Ray       Increased Odds
          of Classical X-Ray \u2192",
            zero=1, cex=0.9, lineheight = unit(6,"mm"), boxsize=0.5,
          colgap=unit(6,"mm"),
            lwd.ci=2, ci.vertices=TRUE, ci.vertices.height = 0.4,
            title="Odds Ratio of Classical COVID-19 Findings on Chest X-Ray",
            txt_gp=fpTxtGp(label=gpar(cex=1.25),
                           ticks=gpar(cex=1.1),
                           xlab=gpar(cex = 1.2),
                           title=gpar(cex = 1.2)),
            graphwidth = unit(200,"mm")

)
```



9.2 Correlation Matrix

9.2.0.0.0.1 This section creates a plot of correlation between all the variables in the raw data

```
library(corrplot)
library(Hmisc)
```

9.2.0.0.0.2 Relevel factors so relevant value is first

```
data$XRPositive <- relevel(data$XRPositive,
    "Negative")

data$Admitted <- relevel(data$Admitted, "Discharged")
data$AdmittedToITU <- relevel(data$AdmittedToITU,
    "No")
```

9.2.0.0.0.3 New variable for correlation matrix

```
cor <- data
```

9.2.0.0.0.4 Remove variables with high missings/ data which won't work e.g. date, RT-PCR removed as it only represents initial ED swab, OverallPos used instead as this includes susequent swabs in 30 days

```
cor<-subset(data, select = -c(CT,DateOfDeath,DateOfDischarge,RTPCR,

         DateOfVisit,DateOfSymptomOnset,FollowUpPos,TimeToDeath,NEWS))'
```

9.2.0.0.0.5 Format and re-name values

```
cor$CTPositive <- ifelse(cor$CTBSTI == "1",
    "Positive", "Negative")
cor$CTPositive <- as.factor(cor$CTPositive)
cor$CTPositive <- relevel(cor$CTPositive,
```

```
    "Negative")
cor$Death <- as.factor(ifelse(cor$ThirtyDayFU ==
    "4", "Dead", "Alive"))
cor$Death <- relevel(cor$Death, "Alive")
cor$OverallPos <- as.factor(cor$OverallPos)
cor <- sapply(cor, as.numeric)
```

9.2.0.0.0.6 Create new numerical correlation matrix

```
cormatrixall <- cor(cor, method = "spearman",
    use = "pairwise.complete.obs")
```

9.2.0.0.0.7 This variable also contains p-values so identification of only significant correlations is possible:

```
cormatrixall2 <- rcorr(as.matrix(cor), type = "spearman")
```

9.2.0.0.0.8 Function to create and format correlation matrix plot

```
corrplot(cormatrixall2$r, method = "color",
    type = "full", order = "hclust", p.mat = cormatrixall2$p,
    sig.level = 0.05, insig = "blank", tl.col = "black",
    outline = "white", title = "Correlation Matrix of Explanatory and Outcome
        Variables",
    line = -1, cex.main = 2, adj.main = 0.5)
```



9.3 STARD Flow Diagram

9.3.0.0.0.1 See instructions from https://www.r-bloggers.com/flow-charts-in-r/

9.3.0.0.0.2 Produces flow charts in Figure 1, (images need to be stretched out, output as svgs)

```
library(grid)
library(Gmisc)

grid.newpage()
# set some parameters to use repeatedly
leftx <- 0.25
```

9.3 STARD Flow Diagram 47

```
midx <- 0.5
rightx <- 0.75
width <- 0.4
gp <- gpar(fill = "white")
# create boxes
(totalattendance <- boxGrob("Number of Patients Attending Emergency Department
         (ED) in Study Period\n n = 1862",
    x = midx, y = 0.9, box_gp = gp, width = 0.7))

(numberwithxr <- boxGrob("Total Number of Patients with Chest X-ray\n n =
         1772",
    x = midx, y = 0.75, box_gp = gp, width = width))
# connect boxes like this
connectGrob(totalattendance, numberwithxr,
    "v")

(numberwithoutxr <- boxGrob("No Chest X-ray\n n = 90",
    x = rightx, y = 0.825, box_gp = gp, width = unit(2,
        "inch"), height = 0.05))

connectGrob(totalattendance, numberwithoutxr,
    "-")

(XRPos <- boxGrob("Chest X-ray Positive for COVID-19 \n n = 750",
    x = leftx, y = 0.6, box_gp = gp, width = width))
(XRNeg <- boxGrob("Chest X-ray Negative for COVID-19\n n = 1022",
    x = rightx, y = 0.6, box_gp = gp, width = width))

connectGrob(numberwithxr, XRPos, "N")
connectGrob(numberwithxr, XRNeg, "N")

(RTPCRXRPos <- boxGrob("Chest X-Ray Positive with RT-PCR swab\n n = 625",
    x = leftx, y = 0.4, box_gp = gp, width = width))
(RTPCRXRNeg <- boxGrob("Chest X-Ray Negative with RT-PCR swab \n n = 573",
    x = rightx, y = 0.4, box_gp = gp, width = width))

connectGrob(XRPos, RTPCRXRPos, "N")
connectGrob(XRNeg, RTPCRXRNeg, "N")

(NoRTPCRXRPos <- boxGrob("No RT-PCR Swab\n n = 125",
    x = 0.4, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))
(NoRTPCRXRNeg <- boxGrob("No RT-PCR Swab\n n = 449",
    x = 0.9, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))

connectGrob(XRPos, NoRTPCRXRPos, "-")
connectGrob(XRNeg, NoRTPCRXRNeg, "-")

(MatchedXRPos <- boxGrob("Chest X-Ray Positive \nafter Propensity Score
         Matching\n n = 430",
    x = leftx, y = 0.225, box_gp = gp, width = width))
(MatchedXRNeg <- boxGrob("Chest X-Ray Negative \nafter Propensity Score
         Matching \n n = 430",
    x = 0.65, y = 0.25, box_gp = gp, width = unit(4.2,
        "inch")))

connectGrob(RTPCRXRPos, MatchedXRPos, "N")
connectGrob(RTPCRXRNeg, MatchedXRNeg, "N")
```

```
(UnmatchedXRPos <- boxGrob("Unmatched\n n = 195",
    x = 0.4, y = 0.325, box_gp = gp, width = unit(1.5,
        "inch")))
(UnmatchedXRNeg <- boxGrob("Unmatched\n n = 143",
    x = 0.9, y = 0.325, box_gp = gp, width = unit(1.5,
        "inch")))

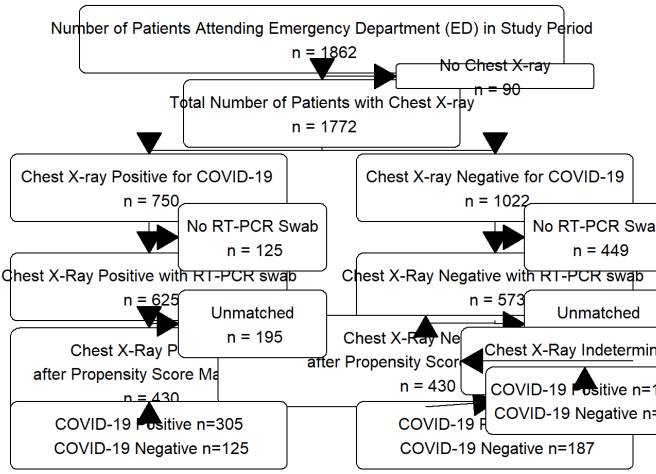
connectGrob(RTPCRXRPos, UnmatchedXRPos, "-")
connectGrob(RTPCRXRNeg, UnmatchedXRNeg, "L")

(DiagXRPositive <- boxGrob("COVID-19 Positive n=305\n COVID-19 Negative n=125",
    x = leftx, y = 0.1, box_gp = gp, width = width))
(DiagXRNegative <- boxGrob("COVID-19 Positive n=243 \n COVID-19 Negative
        n=187",
    x = rightx, y = 0.1, box_gp = gp, width = width))

connectGrob(MatchedXRPos, DiagXRPositive,
    "N")
connectGrob(MatchedXRNeg, DiagXRNegative,
    "vertical")


(XRInd <- boxGrob("Chest X-Ray Indeterminate \n n = 197",
    x = 0.88, y = 0.25, box_gp = gp, width = unit(2.5,
        "inch")))

connectGrob(MatchedXRNeg, XRInd, "horizontal")

(DiagXRInd <- boxGrob("COVID-19 Positive n=136\n COVID-19 Negative n=63",
    x = 0.88, y = 0.17, box_gp = gp, width = unit(2,
        "inch")))
connectGrob(XRInd, DiagXRInd, "vertical")
```

9.3 STARD Flow Diagram 49




```
##### CT Flow Chart#####

grid.newpage()
(totalattendance <- boxGrob("Number of Patients Attending Emergency Department
         (ED) in Study Period\n n = 1862",
    x = midx, y = 0.9, box_gp = gp, width = 0.7))

(numberwithCT <- boxGrob("Total Number with Chest Computed Tompgraphy (CT)\n n
        = 319",
    x = midx, y = 0.75, box_gp = gp, width = width))
connectGrob(totalattendance, numberwithCT,
    "vertical")

(numberwithoutCT <- boxGrob("No Chest CT\n n = 1543",
    x = rightx, y = 0.825, box_gp = gp, width = unit(2,
        "inch"), height = 0.05))

connectGrob(totalattendance, numberwithoutCT,
    "-")


(CTPos <- boxGrob("CT Positive for COVID-19 \n n = 232",
    x = leftx, y = 0.6, box_gp = gp, width = width))
(CTNeg <- boxGrob("CT Negative for COVID-19\n n = 87",
    x = rightx, y = 0.6, box_gp = gp, width = width))

connectGrob(numberwithCT, CTPos, "N")
connectGrob(numberwithCT, CTNeg, "N")

(RTPCRCTPos <- boxGrob("CT Positive with RT-PCR swab\n n = 217",
    x = leftx, y = 0.4, box_gp = gp, width = width))
```

```
(RTPCRCTNeg <- boxGrob("CT Negative with RT-PCR swab \n n = 85",
    x = rightx, y = 0.4, box_gp = gp, width = width))

connectGrob(CTPos, RTPCRCTPos, "N")
connectGrob(CTNeg, RTPCRCTNeg, "N")

(NoRTPCRCTPos <- boxGrob("No RT-PCR Swab\n n = 15",
    x = 0.4, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))
(NoRTPCRCTNeg <- boxGrob("No RT-PCR Swab\n n = 2",
    x = 0.9, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))

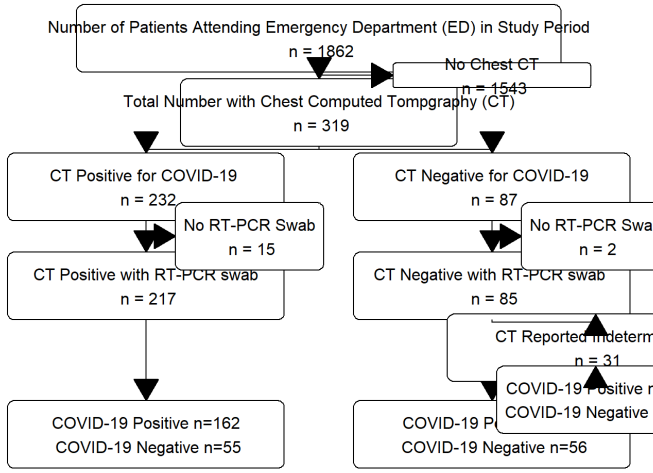
connectGrob(CTPos, NoRTPCRCTPos, "-")
connectGrob(CTNeg, NoRTPCRCTNeg, "-")

(DiagCTPositive <- boxGrob("COVID-19 Positive n=162\n COVID-19 Negative n=55",
    x = leftx, y = 0.1, box_gp = gp, width = width))
(DiagCTNegative <- boxGrob("COVID-19 Positive n=29\n COVID-19 Negative n=56",
    x = rightx, y = 0.1, box_gp = gp, width = width))

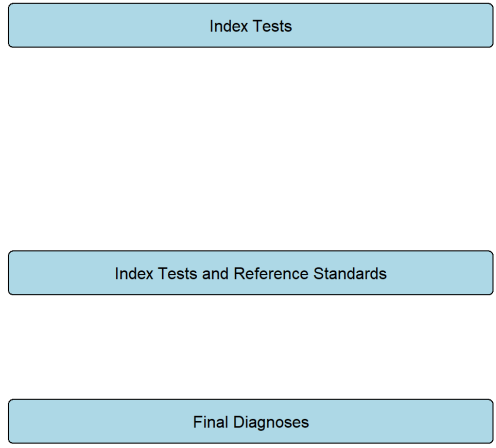
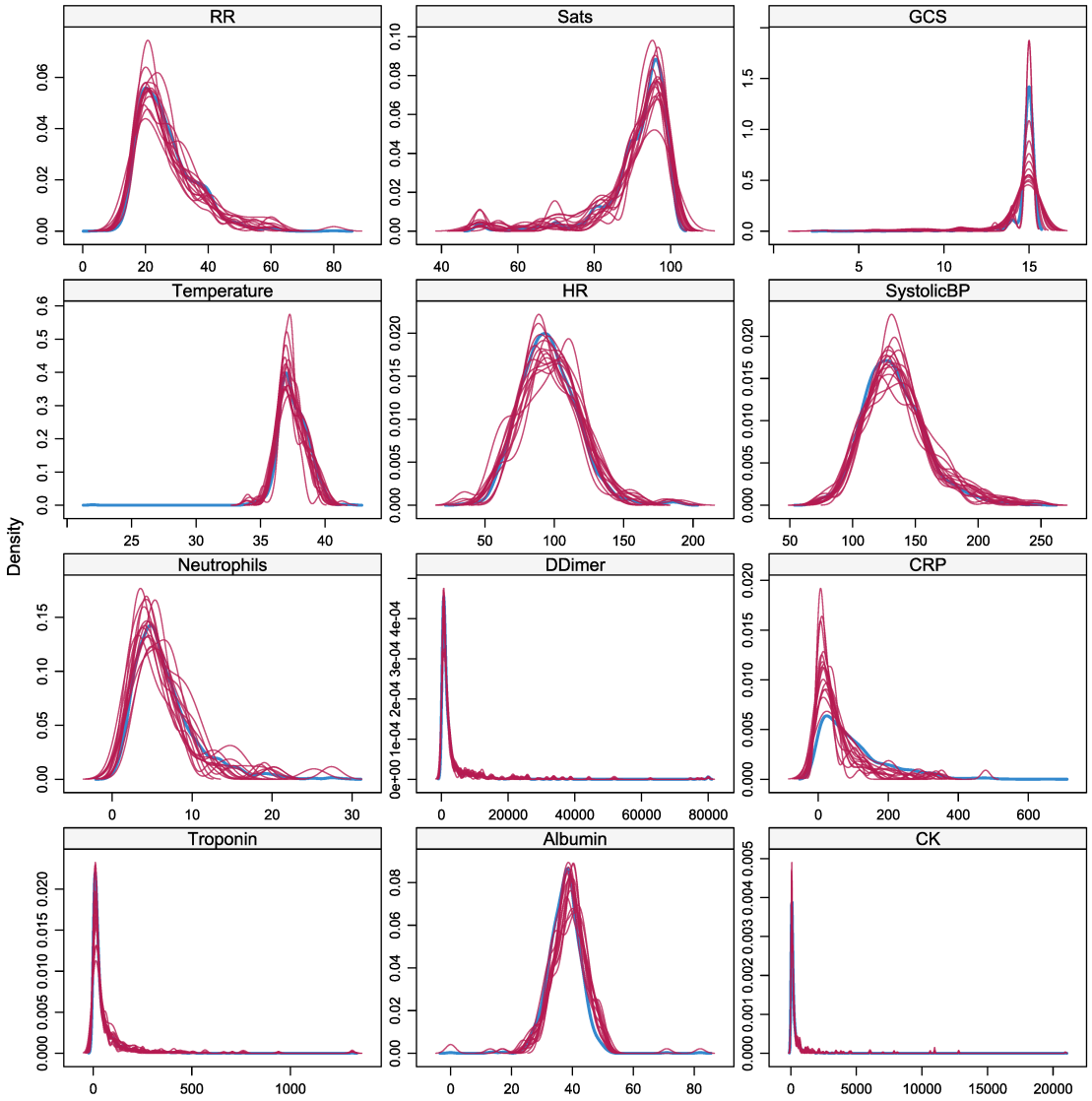
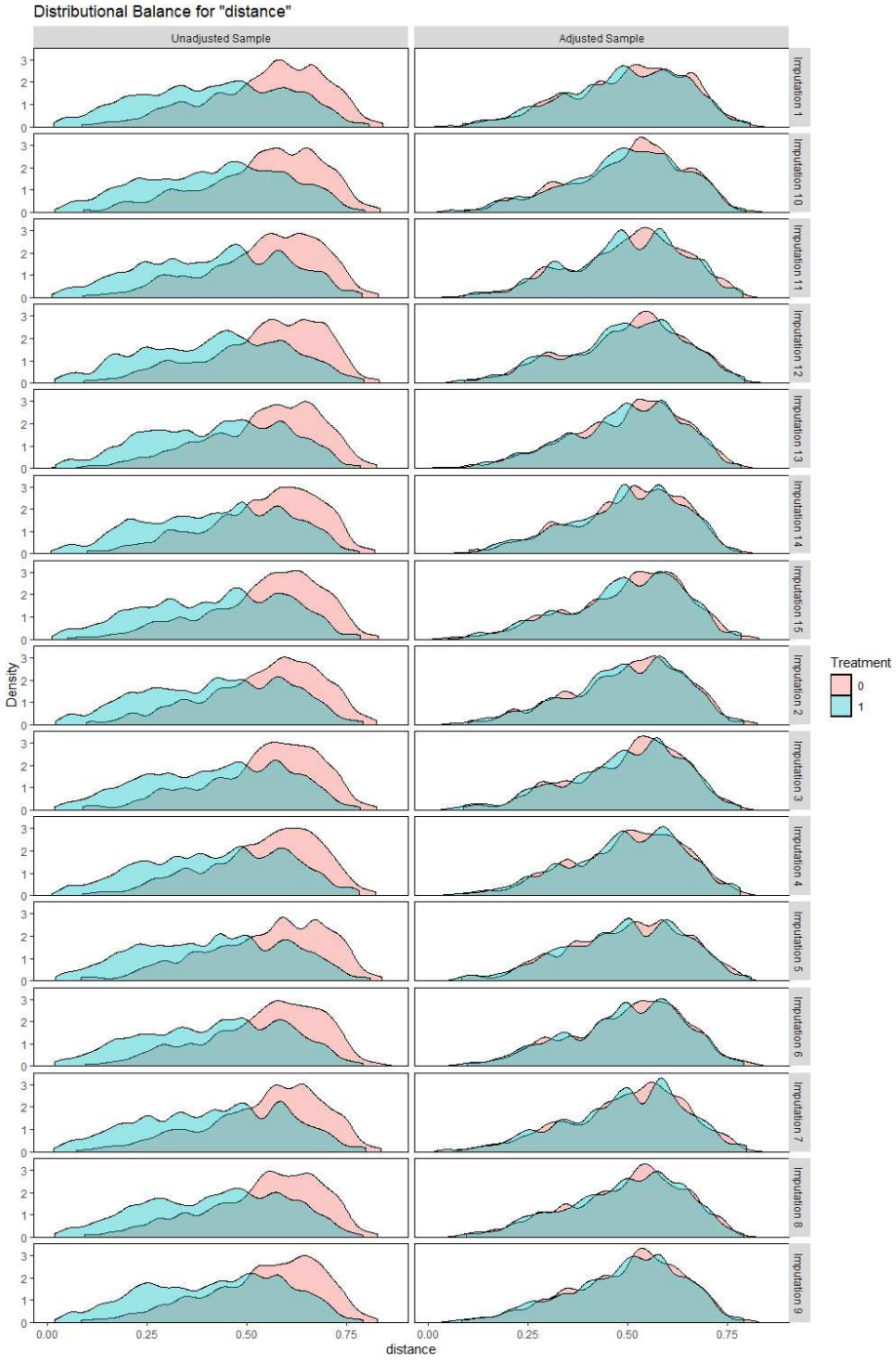
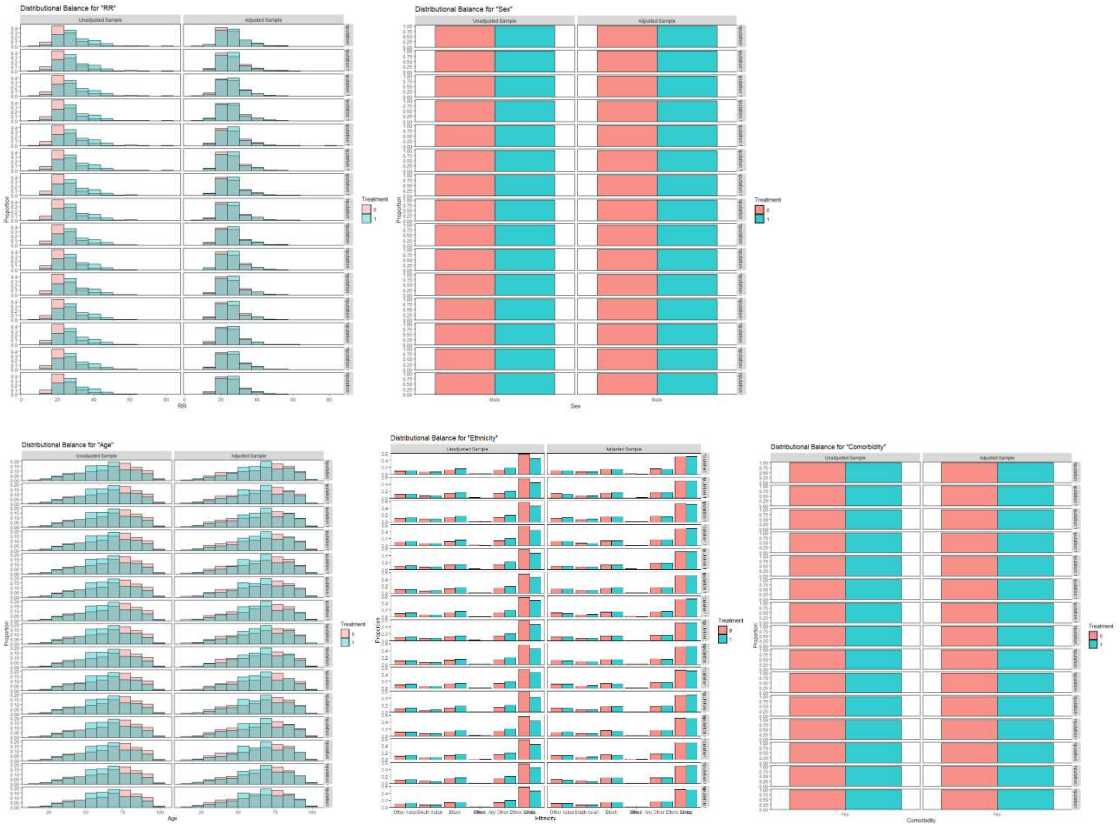
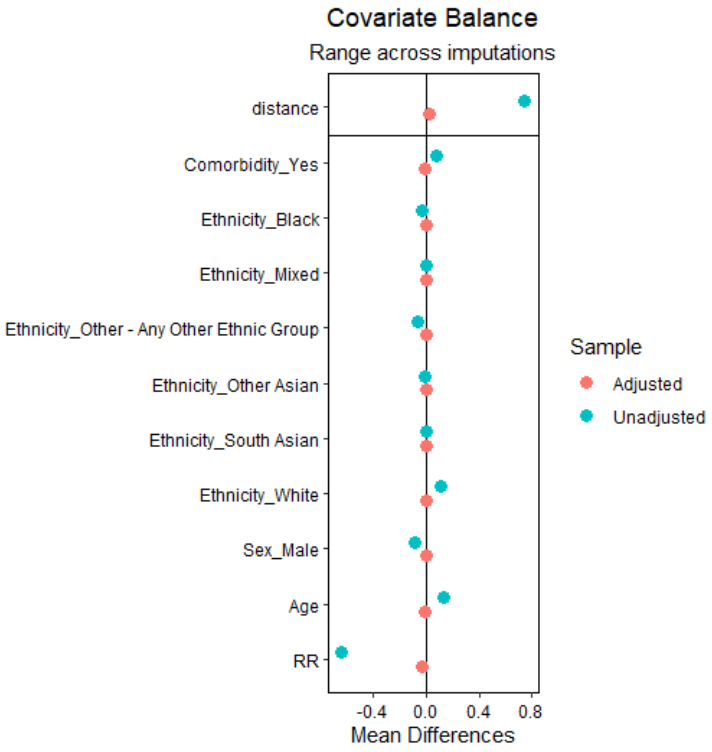
connectGrob(RTPCRCTPos, DiagCTPositive, "N")
connectGrob(RTPCRCTNeg, DiagCTNegative, "N")


(CTInd <- boxGrob("CT Reported Indeterminate \n n = 31",
    x = 0.9, y = 0.275, box_gp = gp, width = unit(3,
        "inch")))

connectGrob(RTPCRCTNeg, CTInd, "N")

(DiagCTInd <- boxGrob("COVID-19 Positive n=16\n COVID-19 Negative n=15",
    x = 0.9, y = 0.17, box_gp = gp, width = unit(2,
        "inch")))
connectGrob(CTInd, DiagCTInd, "vertical")
```

9.3 STARD Flow Diagram 51




```
### Labels####
grid.newpage()
(indextest <- boxGrob("Index Tests", x = midx,
    y = 0.9, box_gp = gpar(fill = "light blue"),
    width = 0.7))

(reftest <- boxGrob("Index Tests and Reference Standards",
    x = midx, y = 0.4, box_gp = gpar(fill = "light blue"),
    width = 0.7))

(finaldiag <- boxGrob("Final Diagnoses",
    x = midx, y = 0.1, box_gp = gpar(fill = "light blue"),
    width = 0.7))
```

52 9 Pooled Regression after Multi…



Supplementary file 1



Supplementary figure 1- Density plots of imputed datasets; Blue represents original dataset; other colours are individual imputed datasets (n=15)

Covariate:	Means Treated	Means Control	Standard Deviation Control	Mean Difference
Overall Propensity Score	0.422997940	0.53935303	0.1449627	-0.1163550897
Female	36.3782051	45.026178	0.4979547	-8.64797288
Male	63.6217949	54.973822	0.4979547	8.64797288
Age	63.796474359	66.19022688	18.5893357	-23.937525171
Comorbidity- Yes	76.1217949	84.467714	0.3625287	-8.34591892
Ethnicity- South Asian	6.5705128	6.631763	0.2490539	-0.06124983
Ethnicity- Black	16.1858974	11.518325	0.3195219	4.66757283
Ethnicity- Mixed	0.9615385	1.396161	0.1174340	-0.43462210
Ethnicity- Other	18.9102564	13.263525	0.3394765	5.64673110
Ethnicity- White	46.6346154	57.766143	0.4943635	-11.13152772
Respiratory Rate	29.214743590	24.01745201	7.2639816	5.1972915828

Supplementary table 1- Means of data before multiple imputation and propensity score matching

	Type	Minimum Difference Adjusted	Mean Difference Adjusted	Maximum Difference Adjusted
Distance	Distance	0.016988	0.027107	0.040963
Sex = Male	Binary	-0.03917	-0.0028	0.015982
Age	Contin.	-0.04586	-0.01371	0.027589
Comorbidity = Yes	Binary	-0.02331	-0.00778	0.004598
Ethnicity = Other Asian	Binary	-0.01392	0.002362	0.016471
Ethnicity = South Asian	Binary	-0.01399	-0.00136	0.011905
Ethnicity = Black	Binary	-0.01852	0.000443	0.015982
Ethnicity = Mixed	Binary	-0.00464	0.001403	0.007042
Ethnicity = Other	Binary	-0.01152	4.30E-06	0.00939
Ethnicity = White	Binary	-0.02353	-0.00285	0.018433
Respiratory Rate	Contin.	-0.06157	-0.03478	-0.00442

Supplementary table 2- Balance summary across imputations

	XR- Negative	XR- Positive	Total
All	573	625	1,198
Matched	430	430	860
Unmatched	143	195	338
Discarded	0	0	0

Supplementary table 3- Average Sample sizes pre- and post- matching across imputed data sets



Supplementary figure 2- Density plot of propensity scores pre- and post- matching in each imputed dataset; treatment units represent a positive X-ray for COVID-19, whereas a control unit represents a negative X-ray



Supplementary figure 3- Histogram of distributions for each matching covariate pre- and post- matching in each imputed dataset; treatment units represent a positive X-ray for COVID-19, whereas a control unit represents a negative X-ray



Supplementary figure 4- Love plot of pooled balances across imputed datasets in matching covariates after matching

CXR in COVID Analysis

Dr Aditya Borakati

Royal Free Hospital, Pond Street, London, NW3 2QG
a.borakati@doctors.org.uk

2020-10-06

/

1 Software Environment and Packages 7
1.1 Load Packages and Data 9
1.1.1 Load Packages: 9
1.2 Power Calculation 9
2 Load Data: 11
3 Data Cleaning 13
3.0.1 Follow Up Swabs + Initial Swabs Positive: 14
3.0.2 Paired XR and RT-PCR data 14
4 Demographic table of raw data 17
5 Imputation 21
6 Propensity Score Matching 23
6.1 Match Balance Diagnostics 23
7 Matched Demographics Table: 25
8 Diagnostic Accuracy 27
8.1 CT Data and Accuracy 28
8.2 CT and XR accuracy comparison 30
8.2.1 Sensitivity 30
8.3 Intermodality Agreement 34
8.3.1 Diagnostic Accuracy Analysis when Indeterminate Reports of CXR and CT are taken as positive 35
9 Pooled Regression after Multiple Imputation and Propensity Score Matching 39
9.0.1 Pooled Univariate Odds Ratios for OverallPos as dependent variable 39
9.0.2 Binomial Logistic Regression with Positive Chest X-ray Report as Dependent Variable 40
9.0.3 Univariate XRPositive as dependent 40
9.0.4 Multivariate XRPositive as dependent 40
9.0.5 Pooled Ordinal Logistic Regression with XRPositive as dependent 41

9.1 Forest Plots 41
9.2 Correlation Matrix 43
9.3 STARD Flow Diagram 46

1 Software Environment and Packages

```
R version 4.0.0 (2020-04-24)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 19041)
Matrix products: default
locale:
LC_COLLATE=English_United Kingdom.1252  LC_CTYPE=English_United Kingdom.1252
LC_MONETARY=English_United Kingdom.1252 LC_NUMERIC=C
LC_TIME=English_United Kingdom.1252
attached base packages:
stats     graphics  grDevices utils     datasets  methods   base
other attached packages:
corrplot 0.84
  Taiyun Wei and Viliam Simko (2017). R package "corrplot": Visualization of
  a Correlation Matrix (Version 0.84). Available from
  https://github.com/taiyun/corrplot
MKmisc 1.6
  Kohl M (2019). MKmisc: Miscellaneous functions from M. Kohl_. R package
        version 1.6, http://www.stamats.de
epiR 1.0-14
  Mark Stevenson with contributions from Telmo Nunes, Cord Heuer, Jonathon
  Marshall, Javier Sanchez, Ron Thornton, Jeno Reiczigel, Jim Robison-Cox,
  Paola Sebastiani, Peter Solymos, Kazuki Yoshida, Geoff Jones, Sarah
  Pirikahu, Simon Firestone, Ryan Kyle, Johann Popp, Mathew Jay and Charles
  Reynard. (2020). epiR: Tools for the Analysis of Epidemiological Data. R
  package version 1.0-14. https://CRAN.R-project.org/package=epiR
Matching 4.9-7
  Jasjeet S. Sekhon (2011). Multivariate and Propensity Score Matching
  Software with Automated Balance Optimization: The Matching Package for R.
  Journal of Statistical Software, 42(7), 1-52. URL
        http://www.jstatsoft.org/v42/i07/.
MASS 7.3-51.5
  Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S.
  Fourth Edition. Springer, New York. ISBN 0-387-95457-0
Ordinal 2019.12-10
  Christensen, R. H. B. (2019). ordinal - Regression Models for Ordinal Data. R
        package version    2019.12-10. https://CRAN.R-
        project.org/package=ordinal.
Hmisc 4.4-0
  Frank E Harrell Jr, with contributions from Charles Dupont and many
  others. (2020). Hmisc: Harrell Miscellaneous. R package version 4.4-0.
  https://CRAN.R-project.org/package=Hmisc
Formula 1.2-3
  Achim Zeileis, Yves Croissant (2010). Extended Model Formulas in R:
  Multiple Parts and Multiple Responses. Journal of Statistical Software
  34(1), 1-13. doi:10.18637/jss.v034.i01
lattice 0.20-41
  Sarkar, Deepayan (2008) Lattice: Multivariate Data Visualization with R.
  Springer, New York. ISBN 978-0-387-75968-5
```

```
mice 3.8.0
  Stef van Buuren, Karin Groothuis-Oudshoorn (2011). mice: Multivariate
  Imputation by Chained Equations in R. Journal of Statistical Software,
  45(3), 1-67. URL https://www.jstatsoft.org/v45/i03/.
readxl 1.3.1
  Hadley Wickham and Jennifer Bryan (2019). readxl: Read Excel Files. R
  package version 1.3.1. https://CRAN.R-project.org/package=readxl
finalfit 1.0.1
  Ewen Harrison, Tom Drake and Riinu Ots (2020). finalfit: Quickly Create
  Elegant Regression Results Tables and Plots when Modelling. R package
  version 1.0.1. https://CRAN.R-project.org/package=finalfit
MatchIt 3.0.2
  Daniel E. Ho, Kosuke Imai, Gary King, Elizabeth A. Stuart (2011). MatchIt:
  Nonparametric Preprocessing for Parametric Causal Inference. Journal of
  Statistical Software, Vol. 42, No. 8, pp. 1-28. URL
  http://www.jstatsoft.org/v42/i08/
tableone 0.11.1
  Kazuki Yoshida (2020). tableone: Create 'Table 1' to Describe Baseline
  Characteristics. R package version 0.11.1.
  https://CRAN.R-project.org/package=tableone
forcats 0.5.0
  Hadley Wickham (2020). forcats: Tools for Working with Categorical
  Variables (Factors). R package version 0.5.0.
  https://CRAN.R-project.org/package=forcats
stringr 1.4.0
  Hadley Wickham (2019). stringr: Simple, Consistent Wrappers for Common
  String Operations. R package version 1.4.0.
  https://CRAN.R-project.org/package=stringr
dplyr 0.8.5
  Hadley Wickham, Romain François, Lionel Henry and Kirill Müller (2020).
  dplyr: A Grammar of Data Manipulation. R package version 0.8.5.
  https://CRAN.R-project.org/package=dplyr
purrr 0.3.4
  Lionel Henry and Hadley Wickham (2020). purrr: Functional Programming
  Tools. R package version 0.3.4. https://CRAN.R-project.org/package=purrr
readr 1.3.1
  Hadley Wickham, Jim Hester and Romain Francois (2018). readr: Read
  Rectangular Text Data. R package version 1.3.1.
  https://CRAN.R-project.org/package=readr
tidyr 1.0.2
  Hadley Wickham and Lionel Henry (2020). tidyr: Tidy Messy Data. R package
  version 1.0.2. https://CRAN.R-project.org/package=tidyr
tibble 3.0.0
  Hadley Wickham and Lionel Henry (2020). tidyr: Tidy Messy Data. R package
  version 1.0.2. https://CRAN.R-project.org/package=tidyr
ggplot2 3.3.0
  H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag
  New York, 2016.
tidyverse 1.3.0
  Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source
  Software, 4(43), 1686, https://doi.org/10.21105/joss.01686
forestplot 1.9
  Max Gordon and Thomas Lumley (2019). forestplot: Advanced Forest Plot Using
         'grid' Graphics. R package version 1.9.   https://CRAN.R-
         project.org/package=forestplot
MatchThem 0.9.3
  Farhad Pishgar and Noah Greifer (2020). MatchThem: Matching and Weighting
         Multiply Imputed Datasets. R package version 0.9.3. https://CRAN.R-
         project.org/package=MatchThem
```

```
miceadds 3.9-14
 Robitzsch, A., & Grund, S. (2020). miceadds: Some Additional Multiple
         Imputation Functions, Especially for 'mice'. R package version 3.9-14.
         https://CRAN.R-project.org/package=miceadds
cobalt 4.2.2
 Noah Greifer (2020). cobalt: Covariate Balance Tables and Plots. R package
         version 4.2.2.   https://CRAN.R-project.org/package=cobalt
```

1.1 Load Packages and Data

1.1.1 Load Packages:

```
library(MKmisc)
library(tidyverse)
library(tableone)
library(MatchIt)
library(finalfit)
library(readxl)
library(cobalt)
library(mice)
library(miceadds)
library(Hmisc)
library(epiR)
library(MatchThem)
library(ordinal)
library(forestplot)
```

1.2 Power Calculation

1.2.0.0.0.1 This code calculates the sample size (positive and negative by gold standard test) needed to evaluate a diagnostic test with 56% sensitivity at 80% power with alpha 0.05. The 56% value is the lower confidence reported by Wong et al. and lower sensitivities typically require higher sample sizes, the result is the same whether specificity or sensitivities are passed as arguments, the previously published specificities are higher than sensitivities so for a generous estimate, the sensitivity was used.

```
power <- power.diagnostic.test(sens = 0.56,
    sig.level = 0.05, delta = 0.1, power = 0.8) %>%
    print()
```

```
 Diagnostic test exact power calculation

           sens = 0.56
              n = 165
             n1 = 165
          delta = 0.1
      sig.level = 0.05
          power = 0.8
           prev = NULL

NOTE: n is number of cases, n1 is number of controls
```

2 Load Data:

```
data <- read_csv("FullDataWithCT.csv", col_types = cols(Age = col_integer(),
    Albumin = col_number(), CK = col_number(),
    CT = col_character(), CRP = col_number(),
    DDimer = col_number(), DateOfDeath = col_date(format = "%d/%m/%Y"),
    DateOfDischarge = col_date(format = "%d/%m/%Y"),
    DateOfVisit = col_date(format = "%d/%m/%Y"),
    DateOfSymptomOnset = col_date(format = "%d/%m/%Y"),
    DiastolicBP = col_number(), FiO2 = col_skip(),
    GCS = col_number(), HR = col_number(),
    MRN = col_skip(), NEWS = col_number(),
    `NEWS2(noFiO2)` = col_skip(), Neutrophils = col_number(),
    RR = col_number(), Sats = col_number(),
    `Supplemental Oxygen` = col_skip(), SystolicBP = col_number(),
    Temperature = col_number(), Troponin = col_number(),
    CTBSTI = col_integer()))
```

3 Data Cleaning

3.0.0.0.0.1 Format data into factors/ differences between dates:

```
data <- mutate_if(data, is.character, as.factor)
data$DayOfSymptoms <- difftime(data$DateOfVisit,
    data$DateOfSymptomOnset, units = "days")
data$TimeToDeath <- abs(difftime(data$DateOfDeath,
    data$DateOfVisit, units = "days"))
data$DayOfSymptoms <- as.numeric(data$DayOfSymptoms)
data$TimeToDeath <- as.numeric(data$TimeToDeath)
```

3.0.0.1 Recode ethnicities as too many options:

3.0.0.1.0.1 This code collapses the ethnicity categories into 'White', 'Black', 'South Asian', 'Other Asian', 'Mixed' or 'Other';

```
data$Ethnicity <- fct_collapse(data$Ethnicity,
    White = c("White - British", "White - Irish",
        "White - Any Other White Background"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    Black = c("Black - Any Other Black Background",
        "Black or Black British - A0rican",
        "Black or Black British - African",
        "Black or Black British - Caribbean"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    `South Asian` = c("Asian or Asian British - Bangladeshi",
        "Asian or Asian British - Indian",
        "Asian or Asian British - Pakistani"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    `Other Asian` = c("Asian - Any Other Asian Background",
        "Other - Chinese"))
data$Ethnicity <- fct_collapse(data$Ethnicity,
    Mixed = c("mixed - Any Other mixed Background",
        "Mixed - Any Other Mixed Background",
        "Mixed - White and Asian", "Mixed - White and Black African",
        "mixed - White and Black Caribbean",
        "Mixed - White and Black Caribbean"))
```

3.0.0.1.0.2 New XR positive column for "Classic Covid" or not:

```
data$XRPositive <- ifelse(data$XRChest ==
    "Classic COVID", "Positive", "Negative")
data$XRPositive <- as.factor(data$XRPositive)
```

3.0.1 Follow Up Swabs + Initial Swabs Positive:

3.0.1.0.0.1 Creates new column 'OverallPos' which includes initial RT-PCR swab and follow-up swabs in 30 days of attendance, if any are positive the value will be positive in this column

```
data$OverallPos <- case_when(data$RTPCR ==
    "Positive" | data$FollowUpPos == "Positive" ~
    "Positive")
data$OverallPos <- replace_na(data$OverallPos,
    "Negative")
```

3.0.1.0.0.2 Create new vector with all variable names (i.e. the column headers)

```
explanatory <- names(data)
```

3.0.2 Paired XR and RT-PCR data

3.0.2.1 Creates new variable 'completedata' which contains only patients who had both CXR and RT-PCR in ED

```
completedata <- filter(data, !is.na(data$XRPositive) &
    !is.na(data$RTPCR))
```

3.0.2.1.1 Remove missing data variable

```
completedata <- completedata[-c(31)]
```

3.0.2.2 Format complete data variables

```
completedata$OverallPos <- as.factor(completedata$OverallPos)

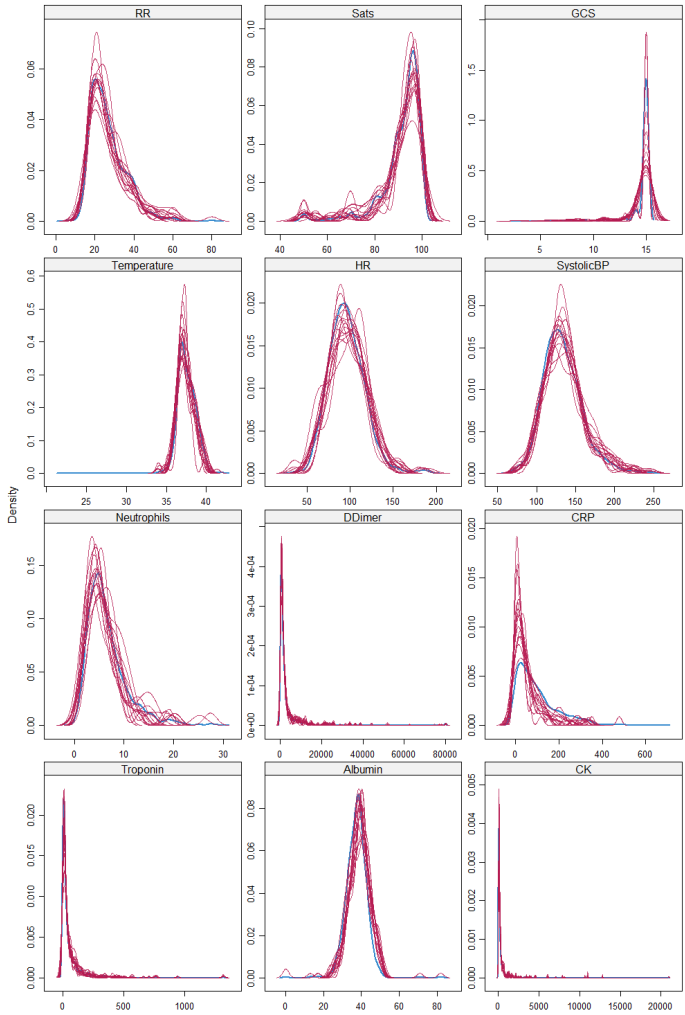
completedata$ThirtyDayFU <- as.factor(completedata$ThirtyDayFU)
completedata$TimeToDeath <- abs(difftime(completedata$DateOfDeath,
    completedata$DateOfVisit, units = "days"))

completedata$TimeToDeath <- as.numeric(completedata$TimeToDeath)
```

3.0.2.2.0.1 Set 'XRChest' as ordinal variable on scale of 'Alternative pathology' as lowest value and 'Classical COVID' as highest

```
completedata$XRChest <- ordered(completedata$XRChest,
    levels = c("Alternative pathology", "No abnormalities",
        "Indeterminate", "Classic COVID"))
```

3.0.2.2.0.2 Convert CT BSTI grade column into factor:

```
completedata$CTBSTI <- as.factor(completedata$CTBSTI)
```

4 Demographic table of raw data

4.0.0.0.0.1 This code creates an unformatted demographic table (table 2 in manuscript), for the raw data, stratified by RT-PCR status, significance testing between RT-PCR +ve and -ve groups is carried out automatically using chi squared, t-tests, ANOVA etc.; there is also a column for the proportion of missing data

```
CreateTableOne(vars = explanatory,
               strata = 'OverallPos',
               data = completedata) -> demogtable

#### List nonnormal factors for summarisation as median / IQR and non
        parametric statistical test

explanatorynnormal<-c("Sats","RR", "GCS", "SystolicBP", "Temperature", "HR",
        "Neutrophils",
+                      "DDimer","Albumin","CRP","CK","Troponin")
as.data.frame(print(demogtable, nonnormal = explanatorynnormal, missing =
        TRUE))->demogtable

write.csv(demogtable, file = "Demogtable.csv")

  Age (mean (SD))                    62.74 (17.72)             66.18 (17.58)
        0.001
  Ethnicity (%)
        0.097
     Other Asian                        29 ( 8.0)                 72 ( 11.8)
     South Asian                        27 ( 7.5)                 38 (  6.2)
     Black                              41 (11.4)                 91 ( 14.9)
     Mixed                               6 ( 1.7)                  6 (  1.0)
     Other - Any Other Ethnic Group     56 (15.5)                105 ( 17.2)
     White                             202 (56.0)                297 ( 48.8)
  Sex = Male (%)                       233 (53.6)                480 ( 62.9)
        0.002
  Sats (median [IQR])                95.00 [92.00, 98.00]      93.00 [88.00,
        96.00]    <0.001 nonnorm
  RR (median [IQR])                  22.00 [20.00, 28.00]      26.00 [20.00,
        32.00]    <0.001 nonnorm
  GCS (median [IQR])                 15.00 [15.00, 15.00]      15.00 [15.00,
        15.00]     0.043 nonnorm
  SystolicBP (median [IQR])         134.00 [119.00, 151.50]   130.00 [115.00,
        145.00]   0.009 nonnorm
  DiastolicBP (mean (SD))            79.54 (16.40)             75.61 (14.51)
        <0.001
  HR (median [IQR])                  96.00 [83.00, 110.00]     94.00 [81.00,
        108.00]    0.092 nonnorm
```

18 4 Demographic table of raw data

Temperature (median [IQR])	37.10 [36.60, 38.00]	37.70 [37.00, 38.40]	<0.001 nonnorm
XRChest (%)			<0.001
Alternative pathology	4 (0.9)	3 (0.4)	
No abnormalities	178 (40.9)	136 (17.8)	
Indeterminate	83 (19.1)	169 (22.1)	
Classic COVID	170 (39.1)	455 (59.6)	
CTPA = **PE** (%)	16 (30.2)	28 (45.9)	0.127
Comorbidity = **Yes** (%)	297 (79.0)	482 (80.3)	0.669
Dyspnoea = **Yes** (%)	274 (69.4)	497 (75.5)	0.034
Neutrophils (median [IQR])	6.42 [4.55, 9.11]	5.25 [3.69, 7.61]	<0.001 nonnorm
DDimer (median [IQR])	1250.00 [619.00, 3059.00]	1105.00 [626.00, 2428.50]	0.204 nonnorm
Albumin (median [IQR])	39.00 [35.00, 42.00]	37.00 [34.00, 40.00]	<0.001 nonnorm
CRP (median [IQR])	51.00 [13.00, 117.00]	83.00 [42.00, 158.00]	<0.001 nonnorm
CK (median [IQR])	91.00 [54.00, 169.00]	146.50 [78.00, 342.75]	<0.001 nonnorm
Troponin (median [IQR])	19.00 [7.00, 53.00]	20.00 [9.00, 53.00]	0.278 nonnorm
Admitted = **Discharged** (%)	104 (24.0)	128 (16.8)	0.003
AdmittedToITU = **Yes** (%)	5 (1.3)	32 (4.8)	0.005
RTPCR = **Positive** (%)	0 (0.0)	738 (96.7)	<0.001
CT = 1 (%)	37 (57.8)	26 (86.7)	0.011
NEWS (**mean** (SD))	4.36 (3.06)	5.48 (2.71)	0.032
ThirtyDayFU (%)			<0.001
1	219 (78.2)	367 (58.3)	
2	14 (5.0)	49 (7.8)	
3	18 (6.4)	60 (9.5)	
4	29 (10.4)	154 (24.4)	
CTBSTI (%)			<0.001
0	23 (22.1)	6 (3.3)	
1	52 (50.0)	157 (85.8)	
2	14 (13.5)	14 (7.7)	
3	15 (14.4)	6 (3.3)	
DayOfSymptoms (**mean** (SD))	9.84 (9.63)	8.56 (15.80)	0.368
TimeToDeath (**mean** (SD))	50.33 (77.93)	57.76 (70.02)	0.618
XRPositive = **Positive** (%)	170 (39.1)	455 (59.6)	<0.001
OverallPos = **Positive** (%)	0 (0.0)	763 (100.0)	

`

4.0.0.0.0.2 Limited dataset comprising relevant data and those without significant missingness:

```
limcompletedata <- dplyr::select(completedata,
    c("Age", "XRChest", "Ethnicity", "Sex",
        "RR", "Sats", "GCS", "Temperature",
        "HR", "SystolicBP", "DiastolicBP",
        "Neutrophils", "DDimer", "CRP", "Troponin",
        "Albumin", "CK", "OverallPos", "Admitted",
        "AdmittedToITU", "ThirtyDayFU", "Dyspnoea",
        "Comorbidity", "XRPositive"))
```

5 Imputation

5.0.0.0.0.1 This code generates 15 imputed datasets using the permuted mean matching method, based on the 'limcompletedata' dataset which has filtered the most relevant fields, with minimal missing data initially

```
imputed <- mice(limcompletedata, m = 15,
    method = "pmm")
```

5.0.0.0.0.2 Imputation Diagnostics Density plot, this corresponds to supplementary figure 1:

```
densityplot(imputed)
```


RR
Sats
GCS
Temperature
HR
SystolicBP
Neutrophils
DDimer
CRP
Troponin
Albumin
CK
Density

6 Propensity Score Matching

6.0.0.0.0.1 This code matches data in the imputed datasets on whether the XR was reported classical COVID or not, the matching is done based on the covariates Sex, Age, Comorbidity, Ethnicity and Respiratory Rate

```
library(MatchThem)
#### MatchThem package requires dependent variable to be coded as 0 or 1
imputed[["data"]][["XRPositive"]] %>% recode_factor("Positive" = "1",
         "Negative" = "0") ->imputed[["data"]][["XRPositive"]]
matchthem(
  XRPositive ~ Sex + Age + Comorbidity + Ethnicity + RR,
  data = imputed,
  method = 'nearest',
  verbose = FALSE,
  replace = FALSE,
  ratio = 1,
  caliper = 0.2,
  m.order = "random",) -> matchedtest
### Set XRChest to unordered for binomial analyses
matchedtest[["datasets"]]c(1:15)[["XRChest"]] %>% factor(ordered = FALSE) ->
         matched2[["datasets"]]c(1:15)[["XRChest"]]
```

6.1 Match Balance Diagnostics

6.1.0.0.1 Creates plots and table with mean difference and distributation of values in covariates betweeen XR +ve and -ve groups after matching across all imputed datasets:

```
#### Supplementary tables 1,2 and 3:
bal.tab(matchedtest)
#### Supplementary figure 2
bal.plot(matchedtest)
#### Supplementary figure 3:
bal.plot(matchedtest, var.name = "Age", type = "histogram",
    which = "both")
bal.plot(matchedtest, var.name = "Sex", type = "histogram",
    which = "both")
bal.plot(matchedtest, var.name = "Ethnicity",
```

```
    type = "histogram", which = "both")
bal.plot(matchedtest, var.name = "RR", type = "histogram",
    which = "both")
bal.plot(matchedtest, var.name = "Comorbidity",
    type = "histogram", which = "both")
#### Supplementary figure 4:
love.plot(matchedtest)
```

7 Matched Demographics Table:

7.0.0.0.0.1 Stack matched imputed datasets into one large datset and split into COVID +ve and -ve groups:

```
### 'all=FALSE' gets matched data only
stacked <- MatchThem::complete(matchedtest,
    n = c(1:15), all = FALSE)
stacked <- stacked %>% filter(.imp > 0)
```

7.0.0.0.0.2 Creates demographics table as above, but on propensity matched imputed datasets, corresponds to Table 4:

```
table4 <- CreateTableOne(strata = "OverallPos",
    data = stacked)
#### Means and SD kept as is, mean counts
#### calculated after dividing by 15 (as 15
#### imputed datasets)
```

7.0.0.0.0.3 Creates demographic table stratified by XR Positive or Negative on matched imputed datasets, correpsonds to Table 5:

```
table5 <- CreateTableOne(strata = "XRPositive",
    data = stacked)
#### Means and SD kept as is, mean counts
#### calculated after dividing by 15 (as 15
#### imputed datasets)
```

7.0.0.0.0.4 Summary statistics for pooled data:

```
### Normal means sd
explanatorynorm <- c("Age", "Temperature",
    "HR", "SystolicBP")
summarynormalOverallPos <- stacked %>% group_by(OverallPos) %>%
```

26 7 Matched Demographics Table:

```
    summarise_at(vars(explanatorynorm), list(mean.default,
        sd))
summarynormalXRPositive <- stacked %>% group_by(XRPositive) %>%
    summarise_at(vars(explanatorynorm), list(mean.default,
        sd))

### Non normal medians and IQR
summarynnormalOverallPos <- stacked %>% group_by(OverallPos) %>%
    summarise_at(vars(explanatorynnormal),
        list(median, IQR))
summarynnormalXRPositive <- stacked %>% group_by(XRPositive) %>%
    summarise_at(vars(explanatorynnormal),
        list(median, IQR))
```

8 Diagnostic Accuracy

8.0.0.1 This section generates the diagnostic accuracy statistics (e.g. sensitivity, specificity) for CXR and CT with RT-PCR as the reference standard using the matched imputed datasets

8.0.0.2 This code creates a contingency table of False/ True Positives and Negatives for Chest X-ray taken from the demographic tables above:

```
contingxr <- matrix(c(305, 243, 125, 187),
    nrow = 2, ncol = 2)

colnames(contingxr) <- c("PCR+", "PCR-")

rownames(contingxr) <- c("XR+", "XR-")
```

8.0.0.2.1 This function calculates diagnostic accuracy test statistics:

```
xraccuracy <- epi.tests(contingxr, conf.level = 0.95)
```

8.0.0.3 Giving the diagnostic accuracy output for CXR in table 3:

```
xraccuracy
          Outcome +    Outcome -      Total
Test +          305          125        430
Test -          243          187        430
Total           548          312        860

Point estimates and 95 % CIs:
-------------------------------------------------------------
Apparent prevalence                   0.50 (0.47, 0.53)
True prevalence                       0.64 (0.60, 0.67)
```

```
Sensitivity                          0.56 (0.51, 0.60)
Specificity                          0.60 (0.54, 0.65)
Positive predictive value            0.71 (0.66, 0.75)
Negative predictive value            0.43 (0.39, 0.48)
Positive likelihood ratio            1.39 (1.19, 1.62)
Negative likelihood ratio            0.74 (0.65, 0.84)
-------------------------------------------------------```
```

8.0.0.3.0.1 NB diagnostic accuracy values in table available in list view of xraccuracy variable

---

8.1 CT Data and Accuracy

8.1.0.0.0.1 Only those with CT and RT PCR:

```
CTdata <- filter(data, is.na(data$CTBSTI) ==
    FALSE & is.na(data$RTPCR) == FALSE)
```

8.1.0.0.0.2 Select relevant variables

```
CTdata <- dplyr::select(CTdata, c("Age",
    "XRChest", "Ethnicity", "Sex", "RR",
    "Sats", "GCS", "Temperature", "HR", "SystolicBP",
    "DiastolicBP", "Neutrophils", "DDimer",
    "CRP", "Troponin", "OverallPos", "Admitted",
    "AdmittedToITU", "ThirtyDayFU", "Dyspnoea",
    "Comorbidity", "XRPositive", "OverallPos",
    "CTBSTI"))
```

8.1.0.0.0.3 Set RT-PCR as factor:

```
CTdata$OverallPos <- as.factor(CTdata$OverallPos)
```

8.1.0.0.0.4 Rename 1 and 0 to Positive and Negative:

```
CTdata$CTPositive <- ifelse(CTdata$CTBSTI ==
    "1", "Positive", "Negative")
CTdata$CTPositive <- as.factor(CTdata$CTPositive)
```

8.1.0.0.0.5 Regression with CT as outcome variable:

```
CT <- finalfit(
  CTdata,
  "OverallPos",
  c(
    "Age",
    "Sex",
    "RR",
    "GCS",
    "CTPositive",
    "Temperature",
    "HR",
    "SystolicBP",
    "DiastolicBP",
    "Sats",
    "Dyspnoea",
    "Comorbidity"

  ),
  confint_level = 0.95
)
```

8.1.0.0.0.6 Contingency table of True/False Positives and Negatives for CT taken from Regression table:

```
contingct <- matrix(c(CT[7, 4], CT[6, 4],
    CT[7, 3], CT[6, 3]), nrow = 2, ncol = 2)
colnames(contingct) <- c("PCR+", "PCR-")
rownames(contingct) <- c("CT+", "CT-")
contingct <- substr(contingct, start = 1,
    stop = 3)
contingct <- sapply(contingct, as.numeric)
contingct <- matrix(contingct, nrow = 2,
    ncol = 2)
colnames(contingct) <- c("PCR+", "PCR-")

rownames(contingct) <- c("CT+", "CT-")
```

8.1.0.0.0.7 Diagnostic accuracy statistics for CT

```
epi.tests(contingct, conf.level = 0.95) -> ctaccuracy
          Outcome +    Outcome -      Total
Test +          162           55        217
Test -           29           56         85
Total           191          111        302

Point estimates and 95 % CIs:
-------------------------------------------------------------
Apparent prevalence                   0.72 (0.66, 0.77)
True prevalence                       0.63 (0.58, 0.69)
Sensitivity                           0.85 (0.79, 0.90)
Specificity                           0.50 (0.41, 0.60)
Positive predictive value             0.75 (0.68, 0.80)
Negative predictive value             0.66 (0.55, 0.76)
Positive likelihood ratio             1.71 (1.41, 2.08)
Negative likelihood ratio             0.30 (0.21, 0.44)
-------------------------------------------------------------
```

8.1.0.0.0.8 NB Diagnostic accuracy values found in list view rather than output

---

8.2 CT and XR accuracy comparison

8.2.0.1 In this section mean differences of diagnostic accuracy statistics between CT and Chest X-ray with confidence intervals and p-values are calculated

8.2.1 Sensitivity

8.2.1.0.0.1 Upper confidence limit for difference in sensitivity

```
ubsens <- (ctaccuracy[["elements"]][["se.up"]] -
    xraccuracy[["elements"]][["se.low"]])
```

8.2.1.0.0.2 Lower confidence limit for difference in sensitivity

```
lbsens <- (ctaccuracy[["elements"]][["se.low"]] -
    xraccuracy[["elements"]][["se.up"]])
```

8.2.1.0.0.3 Mean difference in sensitivity

```
meansens <- ctaccuracy[["elements"]][["se"]] -
    xraccuracy[["elements"]][["se"]]
```

8.2.1.0.0.4 Standard error for sensitivity

```
sesens <- (ubsens - lbsens)/(2 * 1.96)
```

8.2.1.0.0.5 value for difference in sensitivity

```
zsens <- meansens/sesens
```

8.2.1.0.0.6 P-value for difference in sensitivity

```
psens <- exp(-0.717 * zsens - 0.416 * zsens^2)
```

8.2.1.0.0.7 Format values into 'mean difference (95% CI) p-value' rounded to 2 d.p.

```
diffsens <- sprintf("%s (%s-%s)", round(meansens,
    digits = 2), round(lbsens, digits = 2),
    round(ubsens, digits = 2))
diffsensp <- c(diffsens, psens)
```

8.2.1.0.0.8 Subsequent analyses in this section follow the code above

```
## Specificity
ubspec <- (ctaccuracy[["elements"]][["sp.up"]] -
    xraccuracy[["elements"]][["sp.low"]])
lbspec <- (ctaccuracy[["elements"]][["sp.low"]] -
    xraccuracy[["elements"]][["sp.up"]])
meanspec <- ctaccuracy[["elements"]][["sp"]] -
    xraccuracy[["elements"]][["sp"]]
sespec <- (ubspec - lbspec)/(2 * 1.96)
zspec <- meanspec/sespec
pspec <- exp(-0.717 * zspec - 0.416 * zspec^2)
diffspec <- sprintf("%s (%s-%s)", round(meanspec,
    digits = 2), round(lbspec, digits = 2),
    round(ubspec, digits = 2))
diffspecp <- c(diffspec, pspec)


ubda <- (ctaccuracy[["elements"]][["da.up"]] -
    xraccuracy[["elements"]][["da.low"]])
lbda <- (ctaccuracy[["elements"]][["da.low"]] -
    xraccuracy[["elements"]][["da.up"]])
meanda <- ctaccuracy[["elements"]][["da"]] -
    xraccuracy[["elements"]][["da"]]
seda <- (ubda - lbda)/(2 * 1.96)
zda <- meanda/seda
pda <- exp(-0.717 * zda - 0.416 * zda^2)
diffda <- sprintf("%s (%s-%s)", round(meanda,
    digits = 2), round(lbda, digits = 2),
    round(ubda, digits = 2))
diffdap <- c(diffda, pda)
## Positive Likelihood Ratio
ublrpos <- (ctaccuracy[["elements"]][["lrpos.up"]] -
    xraccuracy[["elements"]][["lrpos.low"]])
lblrpos <- (ctaccuracy[["elements"]][["lrpos.low"]] -
    xraccuracy[["elements"]][["lrpos.up"]])
meanlrpos <- ctaccuracy[["elements"]][["lrpos"]] -
    xraccuracy[["elements"]][["lrpos"]]
selrpos <- (ublrpos - lblrpos)/(2 * 1.96)
zlrpos <- meanlrpos/selrpos
plrpos <- exp(-0.717 * zlrpos - 0.416 * zlrpos^2)
difflrpos <- sprintf("%s (%s-%s)", round(meanlrpos,
    digits = 2), round(lblrpos, digits = 2),
```

```
    round(ublrpos, digits = 2))
difflrposp <- c(difflrpos, plrpos)
## Negative Likelihood Ratios
ublrneg <- (ctaccuracy[["elements"]][["lrneg.up"]] -
    xraccuracy[["elements"]][["lrneg.low"]])
lblrneg <- (ctaccuracy[["elements"]][["lrneg.low"]] -
    xraccuracy[["elements"]][["lrneg.up"]])
meanlrneg <- ctaccuracy[["elements"]][["lrneg"]] -
    xraccuracy[["elements"]][["lrneg"]]
selrneg <- (ublrneg - lblrneg)/(2 * 1.96)
zlrneg <- meanlrneg/selrneg
plrneg <- exp(-0.717 * zlrneg - 0.416 * zlrneg^2)
difflrneg <- sprintf("%s (%s-%s)", round(meanlrneg,
    digits = 2), round(lblrneg, digits = 2),
    round(ublrneg, digits = 2))
difflrnegp <- c(difflrneg, plrneg)


## Positive Predictive Value
ppv <- (ctaccuracy[["elements"]][["ppv.low"]] -
    xraccuracy[["elements"]][["ppv.up"]])
meanppv <- ctaccuracy[["elements"]][["ppv"]] -
    xraccuracy[["elements"]][["ppv"]]
seppv <- (ubppv - lbppv)/(2 * 1.96)
zppv <- meanppv/seppv
pppv <- exp(-0.717 * zppv - 0.416 * zppv^2)
diffppv <- sprintf("%s (%s-%s)", round(meanppv,
    digits = 2), round(lbppv, digits = 2),
    round(ubppv, digits = 2))
diffppvp <- c(diffppv, pppv)

## Negative Predictive Value
npv <- (ctaccuracy[["elements"]][["npv.low"]] -
    xraccuracy[["elements"]][["npv.up"]])
meannpv <- ctaccuracy[["elements"]][["npv"]] -
    xraccuracy[["elements"]][["npv"]]
senpv <- (ubnpv - lbnpv)/(2 * 1.96)
znpv <- meannpv/senpv
pnpv <- exp(-0.717 * znpv - 0.416 * znpv^2)
diffnpv <- sprintf("%s (%s-%s)", round(meannpv,
    digits = 2), round(lbnpv, digits = 2),
    round(ubnpv, digits = 2))
diffnpvp <- c(diffnpv, pnpv)


## Apparent Prevalence
meantp <- ctaccuracy[["elements"]][["tp"]] -
    xraccuracy[["elements"]][["tp"]]
setp <- (ubtp - lbtp)/(2 * 1.96)
ztp <- meantp/setp
ptp <- exp(-0.717 * ztp - 0.416 * ztp^2)
difftp <- sprintf("%s (%s-%s)", round(meantp,
    digits = 2), round(lbtp, digits = 2),
    round(ubtp, digits = 2))
difftpp <- c(difftp, ptp)

## True Prevalence
meanap <- ctaccuracy[["elements"]][["ap"]] -
    xraccuracy[["elements"]][["ap"]]
```

```
seap <- (ubap - lbap)/(2 * 1.96)
zap <- meanap/seap
pap <- exp(-0.717 * zap - 0.416 * zap^2)
diffap <- sprintf("%s (%s-%s)", round(meanap,
    digits = 2), round(lbap, digits = 2),
    round(ubap, digits = 2))
diffapp <- c(diffap, pap)
```

8.3 Intermodality Agreement

8.3.0.0.0.1 This section contains code to analyse the level of agreement in the unmatched CT dataset which contains only data with CT, XR and RT-PCR

8.3.0.0.0.2 First- comparing CT and XR agreement

```
library(irr)
kappa2(c(CTdata$XRPositive, CTdata$CTPositive),
    weight = "squared")
d <- CTdata %>% select(c("CTPositive", "XRPositive"))
View(d)
kappa2(d, weight = "squared")
```

8.3.0.0.0.3 Output:

```
 Cohen's Kappa for 2 Raters (Weights: squared)

 Subjects = 287
   Raters = 2
    Kappa = 0.406

        z = 7.14
  p-value = 9.37e-13
```

8.3.0.0.0.4 The following code compares RT-PCR, CT and XR

```
d2 <- CTdata %>% select(c("CTPositive", "XRPositive",
    "OverallPos"))
View(d2)
kappam.fleiss(d2)
```

8.3.0.0.0.5 Output:

```
Fleiss' Kappa for m Raters

Subjects = 287
  Raters = 3
   Kappa = 0.361

       z = 10.6
 p-value = 0
```

8.3.1 Diagnostic Accuracy Analysis when Indeterminate Reports of CXR and CT are taken as positive

8.3.1.1 XR Indeterminates

8.3.1.1.0.1 New column for positive if indeterminate

```
stacked$XRIndPositive <- ifelse(stacked$XRChest ==
    "Classic COVID" | stacked$XRChest ==
    "Indeterminate", "Positive", "Negative")
stacked$XRIndPositive <- as.factor(stacked$XRIndPositive)
stackedpos <- stacked %>% filter(OverallPos ==
    "Positive")
stackedneg <- stacked %>% filter(OverallPos ==
    "Negative")
summary(stackedpos$XRIndPositive)
summary(stackedneg$XRIndPositive)

contingxrind <- matrix(c(441, 107, 186, 126),
    nrow = 2, ncol = 2)
colnames(contingxrind) <- c("PCR+", "PCR-")

rownames(contingxrind) <- c("XR+", "XR-")
xrindaccuracy <- epi.tests(contingxrind)
```

8.3.1.1.0.2 In this section mean differences of diagnostic accuracy statistics between CT (when CT indeterminates are not counted as positive)and Chest X-ray with confidence intervals and p-values are calculated, follows the same pattern as code previously

```
###### Sensitivity Upper confidence limit for
###### difference in sensitivity

ubsens <- (ctaccuracy[["elements"]][["se.up"]] -
    xrindaccuracy[["elements"]][["se.low"]])
## Lower confidence limit for difference
## in sensitivity
lbsens <- (ctaccuracy[["elements"]][["se.low"]] -
    xrindaccuracy[["elements"]][["se.up"]])
## Mean difference in sensitivity
meansens <- ctaccuracy[["elements"]][["se"]] -
    xrindaccuracy[["elements"]][["se"]]
## Standard error for sensitivity
sesens <- (ubsens - lbsens)/(2 * 1.96)
## Z value for difference in sensitivity
zsens <- meansens/sesens
## P-value for difference in sensitivity
psens <- exp(-0.717 * zsens - 0.416 * zsens^2)
### Format values into 'mean difference
### (95% CI) p-value' rounded to 2 d.p.
diffsens <- sprintf("%s (%s-%s)", round(meansens,
    digits = 2), round(lbsens, digits = 2),
    round(ubsens, digits = 2))
diffsensp <- c(diffsens, psens)

### Subsequent analyses in this section
### follow the code above Specificity
ubspec <- (ctaccuracy[["elements"]][["sp.up"]] -
    xrindaccuracy[["elements"]][["sp.low"]])
lbspec <- (ctaccuracy[["elements"]][["sp.low"]] -
    xrindaccuracy[["elements"]][["sp.up"]])
meanspec <- ctaccuracy[["elements"]][["sp"]] -
    xrindaccuracy[["elements"]][["sp"]]
sespec <- (ubspec - lbspec)/(2 * 1.96)
zspec <- meanspec/sespec
pspec <- exp(-0.717 * zspec - 0.416 * zspec^2)
diffspec <- sprintf("%s (%s-%s)", round(meanspec,
    digits = 2), round(lbspec, digits = 2),
    round(ubspec, digits = 2))
diffspecp <- c(diffspec, pspec)


ubda <- (ctaccuracy[["elements"]][["da.up"]] -
    xrindaccuracy[["elements"]][["da.low"]])
lbda <- (ctaccuracy[["elements"]][["da.low"]] -
    xrindaccuracy[["elements"]][["da.up"]])
meanda <- ctaccuracy[["elements"]][["da"]] -
    xrindaccuracy[["elements"]][["da"]]
seda <- (ubda - lbda)/(2 * 1.96)
```

```
zda <- meanda/seda
pda <- exp(-0.717 * zda - 0.416 * zda^2)
diffda <- sprintf("%s (%s-%s)", round(meanda,
    digits = 2), round(lbda, digits = 2),
    round(ubda, digits = 2))
diffdap <- c(diffda, pda)
## Positive Likelihood Ratio
ublrpos <- (ctaccuracy[["elements"]][["lrpos.up"]] -
    xrindaccuracy[["elements"]][["lrpos.low"]])
lblrpos <- (ctaccuracy[["elements"]][["lrpos.low"]] -
    xrindaccuracy[["elements"]][["lrpos.up"]])
meanlrpos <- ctaccuracy[["elements"]][["lrpos"]] -
    xrindaccuracy[["elements"]][["lrpos"]]
selrpos <- (ublrpos - lblrpos)/(2 * 1.96)
zlrpos <- meanlrpos/selrpos
plrpos <- exp(-0.717 * zlrpos - 0.416 * zlrpos^2)
difflrpos <- sprintf("%s (%s-%s)", round(meanlrpos,
    digits = 2), round(lblrpos, digits = 2),
    round(ublrpos, digits = 2))
difflrposp <- c(difflrpos, plrpos)
## Negative Likelihood Ratios
ublrneg <- (ctaccuracy[["elements"]][["lrneg.up"]] -
    xrindaccuracy[["elements"]][["lrneg.low"]])
lblrneg <- (ctaccuracy[["elements"]][["lrneg.low"]] -
    xrindaccuracy[["elements"]][["lrneg.up"]])
meanlrneg <- ctaccuracy[["elements"]][["lrneg"]] -
    xrindaccuracy[["elements"]][["lrneg"]]
selrneg <- (ublrneg - lblrneg)/(2 * 1.96)
zlrneg <- meanlrneg/selrneg
plrneg <- exp(-0.717 * zlrneg - 0.416 * zlrneg^2)
difflrneg <- sprintf("%s (%s-%s)", round(meanlrneg,
    digits = 2), round(lblrneg, digits = 2),
    round(ublrneg, digits = 2))
difflrnegp <- c(difflrneg, plrneg)


## Positive Predictive Value
ppv <- (ctaccuracy[["elements"]][["ppv.low"]] -
    xrindaccuracy[["elements"]][["ppv.up"]])
meanppv <- ctaccuracy[["elements"]][["ppv"]] -
    xrindaccuracy[["elements"]][["ppv"]]
seppv <- (ubppv - lbppv)/(2 * 1.96)
zppv <- meanppv/seppv
pppv <- exp(-0.717 * zppv - 0.416 * zppv^2)
diffppv <- sprintf("%s (%s-%s)", round(meanppv,
    digits = 2), round(lbppv, digits = 2),
    round(ubppv, digits = 2))
diffppvp <- c(diffppv, pppv)

## Negative Predictive Value
npv <- (ctaccuracy[["elements"]][["npv.low"]] -
    xrindaccuracy[["elements"]][["npv.up"]])
meannpv <- ctaccuracy[["elements"]][["npv"]] -
    xrindaccuracy[["elements"]][["npv"]]
senpv <- (ubnpv - lbnpv)/(2 * 1.96)
znpv <- meannpv/senpv
pnpv <- exp(-0.717 * znpv - 0.416 * znpv^2)
diffnpv <- sprintf("%s (%s-%s)", round(meannpv,
    digits = 2), round(lbnpv, digits = 2),
```

```
    round(ubnpv, digits = 2))
diffnpvp <- c(diffnpv, pnpv)


## True Prevalence
meantp <- ctaccuracy[["elements"]][["tp"]] -
    xrindaccuracy[["elements"]][["tp"]]
setp <- (ubtp - lbtp)/(2 * 1.96)
ztp <- meantp/setp
ptp <- exp(-0.717 * ztp - 0.416 * ztp^2)
difftp <- sprintf("%s (%s-%s)", round(meantp,
    digits = 2), round(lbtp, digits = 2),
    round(ubtp, digits = 2))
difftpp <- c(difftp, ptp)

## Apparent Prevalence
meanap <- ctaccuracy[["elements"]][["ap"]] -
    xrindaccuracy[["elements"]][["ap"]]
seap <- (ubap - lbap)/(2 * 1.96)
zap <- meanap/seap
pap <- exp(-0.717 * zap - 0.416 * zap^2)
diffap <- sprintf("%s (%s-%s)", round(meanap,
    digits = 2), round(lbap, digits = 2),
    round(ubap, digits = 2))
diffapp <- c(diffap, pap)
```

8.3.1.2 CT Indeterminates

8.3.1.2.0.1 New column for positive if indeterminate

```
CTdata$CTIndPositive <- ifelse(CTdata$CTBSTI ==
    "1" | CTdata$CTBSTI == "2", "Positive",
    "Negative")
CTdata$CTIndPositive <- as.factor(CTdata$CTIndPositive)
valuesctind <- CTdata %>% group_by(OverallPos,
    CTIndPositive) %>% summarise(n = n())
ctcontingind <- matrix(data = c(178, 13,
    70, 41), nrow = 2, ncol = 2)

colnames(ctcontingind) <- c("PCR+ve", "PCR-ve")
rownames(ctcontingind) <- c("CT+ve", "CT-ve")
ctindaccuracy <- epi.tests(ctcontingind)
```

9 Pooled Regression after Multiple Imputation and Propensity Score Matching

9.0.0.0.0.1 Binomnal Logistic regression with RT-PCR as dependent variable

```
overallposmatchimp <- matchedtest %>% with(glm(formula(ff_formula(dependent =
         "OverallPos",
    explanatory = c("Age", "Ethnicity", "Sex",
        "RR", "GCS", "Temperature", "HR",
        "SystolicBP", "Neutrophils", "DDimer",
        "CRP", "Troponin", "Albumin", "CK",
        "Sats", "Admitted", "AdmittedToITU",
        "ThirtyDayFUTwo", "Dyspnoea", "Comorbidity",
        "XRChest"))), family = "binomial"),
    all = FALSE)
P <- overallposmatchimp %>% pool()
multivarpooledoverallpos = P %>% fit2df(estimate_name = "OR (multiple
         imputation)",
    exp = TRUE)
```

9.0.0.0.0.2 'multivarpooledoverallpos' produces multivariate odds ratios for each explanatory variable, corresponding to Table 4

9.0.1 Pooled Univariate Odds Ratios for OverallPos as dependent variable

9.0.1.0.0.1 This code is run with each of the explanatory variables in table 4 as arguments to produce their respective odds Ratios in table 4

```
overallposmatchimpunivar <- matchedtest %>%
    with(glm(formula(ff_formula(dependent = "OverallPos",
```

```
        explanatory = "XRChest")), family = "binomial"))
P <- overallposmatchimpunivar %>% pool()
univarpooledoverallpos = univaroverallpos <- P %>%
    fit2df(estimate_name = "OR (univariate)",
        exp = TRUE)
univaroverallpos
```

9.0.2 Binomial Logistic Regression with Positive Chest X-ray Report as Dependent Variable

9.0.2.0.0.1 This code follows the format above to produce univariate and multivariate odds ratios for each explanatory variable for having a positive XR report

9.0.3 Univariate XRPositive as dependent

9.0.3.0.0.1 (different explanatory variables passed into function to produce Odds ratios for each)

```
XRChestmatchimp <- matchedtest %>% with(glm(formula(ff_formula(dependent =
         "XRPositive",
    explanatory = "Comorbidity")), family = "binomial"))
P <- XRChestmatchimp %>% pool()
multivarpooledXRChest = univarXRChest <- P %>%
    fit2df(estimate_name = "OR (univariate)",
        exp = TRUE)
univarXRChest
```

9.0.4 Multivariate XRPositive as dependent

```
XRChestmatchimp <- matchedtest %>% with(glm(formula(ff_formula(dependent =
         "XRPositive",
    explanatory = c("Age", "OverallPos",
        "Ethnicity", "Sex", "RR", "GCS",
        "Temperature", "HR", "SystolicBP",
        "Neutrophils", "DDimer", "CRP", "Troponin",
        "Albumin", "CK", "Sats", "Admitted",
        "AdmittedToITU", "ThirtyDayFUTwo",
        "Dyspnoea", "Comorbidity"))), family = "binomial"))
P <- XRChestmatchimp %>% pool()
multivarpooledXRChest = multivarXRChest <- P %>%
    fit2df(estimate_name = "OR (multivariate)",
```

```
        exp = TRUE)
multivarXRChest
```

9.0.5 Pooled Ordinal Logistic Regression with XRPositive as dependent

9.0.5.0.0.1 This code also produces multivariate odds ratios for table 5, however, uses ordinal linear regression after the CXR report variable is converted to an ordered categorical variable, with alternative pathology as the lowest and classic covid as the highest value (see table 3)

```
XRChestmatchimpord <- matchedtest %>% with(clm(formula = XRChest ~
    Age + OverallPos + Ethnicity + Sex +
        RR + GCS + Temperature + HR + SystolicBP +
        Neutrophils + DDimer + CRP + Troponin +
        Sats + Admitted + AdmittedToITU +
        ThirtyDayFUTwo + Dyspnoea + Comorbidity))
P <- pool(object = XRChestmatchimpord[["analyses"]])
multivarpooledXRChestord = multivarXRChestord <- P %>%
    fit2df(estimate_name = "OR (multivariate)",
        exp = TRUE)
multivarXRChestord
```

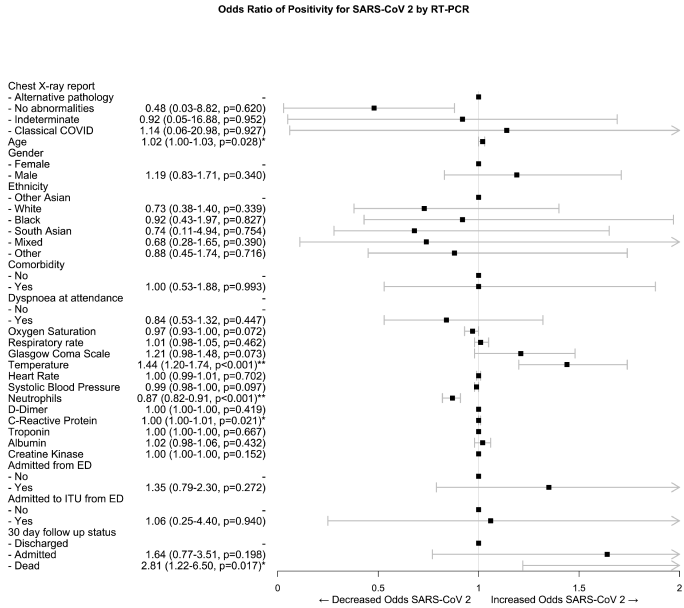
9.1 Forest Plots

9.1.0.0.0.1 Creates forest plots for post matched regression tables above:

```
Figure1Forest <- read_excel("Figure1Forest.xlsx",
    col_types = c("text", "numeric", "numeric",
        "numeric", "text", "text"))

tabletext1 <- cbind(Figure1Forest$explanatory,
    Figure1Forest$summary)
forestplot(tabletext1, Figure1Forest$Mean,
    Figure1Forest$Lower, Figure1Forest$Upper,
    is.summary = FALSE, clip = c(0, 2), xlab = "<U+2190> Decreased Odds SARS-
        CoV 2       Increased Odds SARS-CoV 2 <U+2192>",
    zero = 1, cex = 0.9, lineheight = unit(6,
        "mm"), boxsize = 0.4, colgap = unit(6,
        "mm"), lwd.ci = 2, ci.vertices = TRUE,
    ci.vertices.height = 0.4, title = "Odds Ratio of Positivity for SARS-CoV 2
        by RT-PCR",
    txt_gp = fpTxtGp(label = gpar(cex = 1.25),
        ticks = gpar(cex = 1.1), xlab = gpar(cex = 1.2),
```

```
        title = gpar(cex = 1.2)), graphwidth = unit(200,
        "mm"))
```

9.1.0.0.0.2 Figure 2:

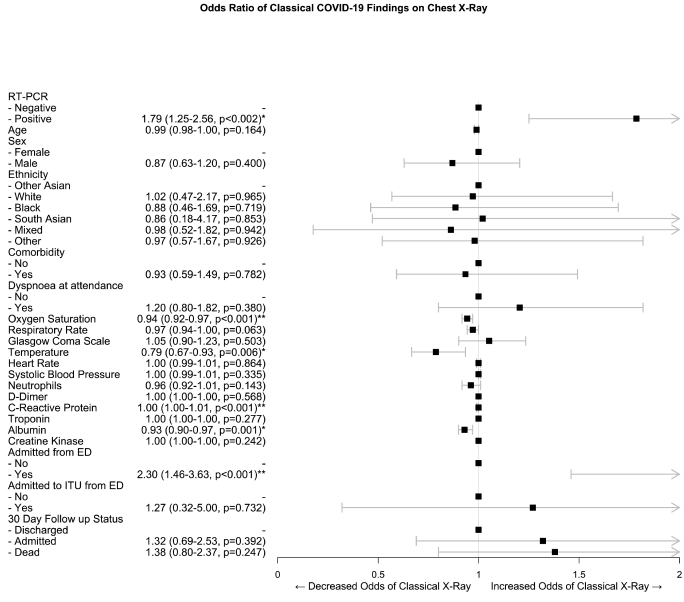


9.1.0.0.0.3 Figure 3 (XR dependent):

```
Figure2Forest <- read_excel("Figure2Forest.xlsx",
                            col_types = c("text", "numeric", "numeric",
                                          "numeric", "text", "text"))

tabletext2<-cbind(Figure2Forest$explanatory,Figure2Forest$summary)
forestplot (tabletext2, Figure2Forest$Mean,
            Figure2Forest$Lower, Figure2Forest$Upper, is.summary = FALSE,
            clip = c(0, 2),
```

9.1 Forest Plots 43

```
          xlab="\u2190 Decreased Odds of Classical X-Ray       Increased Odds
        of Classical X-Ray \u2192",
          zero=1, cex=0.9, lineheight = unit(6,"mm"), boxsize=0.5,
        colgap=unit(6,"mm"),
          lwd.ci=2, ci.vertices=TRUE, ci.vertices.height = 0.4,
          title="Odds Ratio of Classical COVID-19 Findings on Chest X-Ray",
          txt_gp=fpTxtGp(label=gpar(cex=1.25),
                         ticks=gpar(cex=1.1),
                         xlab=gpar(cex = 1.2),
                         title=gpar(cex = 1.2)),
          graphwidth = unit(200,"mm")

)
```



9.2 Correlation Matrix

9.2.0.0.0.1 This section creates a plot of correlation between all the variables in the raw data

```
library(corrplot)
library(Hmisc)
```

9.2.0.0.0.2 Relevel factors so relevant value is first

```
data$XRPositive <- relevel(data$XRPositive,
    "Negative")

data$Admitted <- relevel(data$Admitted, "Discharged")
data$AdmittedToITU <- relevel(data$AdmittedToITU,
    "No")
```

9.2.0.0.0.3 New variable for correlation matrix

```
cor <- data
```

9.2.0.0.0.4 Remove variables with high missings/ data which won't work e.g. date, RT-PCR removed as it only represents initial ED swab, OverallPos used instead as this includes susequent swabs in 30 days

```
cor<-subset(data, select = -c(CT,DateOfDeath,DateOfDischarge,RTPCR,

         DateOfVisit,DateOfSymptomOnset,FollowUpPos,TimeToDeath,NEWS))'
```

9.2.0.0.0.5 Format and re-name values

```
cor$CTPositive <- ifelse(cor$CTBSTI == "1",
    "Positive", "Negative")
cor$CTPositive <- as.factor(cor$CTPositive)
cor$CTPositive <- relevel(cor$CTPositive,
```

```
    "Negative")
cor$Death <- as.factor(ifelse(cor$ThirtyDayFU ==
    "4", "Dead", "Alive"))
cor$Death <- relevel(cor$Death, "Alive")
cor$OverallPos <- as.factor(cor$OverallPos)
cor <- sapply(cor, as.numeric)
```

9.2.0.0.0.6 Create new numerical correlation matrix

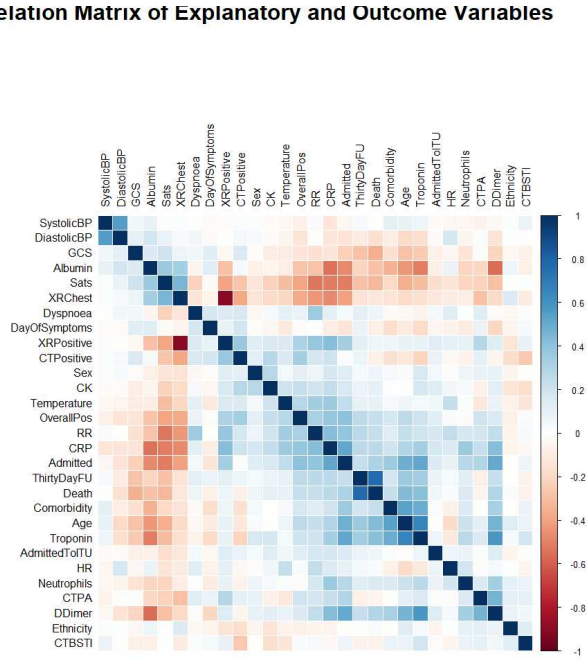
```
cormatrixall <- cor(cor, method = "spearman",
    use = "pairwise.complete.obs")
```

9.2.0.0.0.7 This variable also contains p-values so identification of only significant correlations is possible:

```
cormatrixall2 <- rcorr(as.matrix(cor), type = "spearman")
```

9.2.0.0.0.8 Function to create and format correlation matrix plot

```
corrplot(cormatrixall2$r, method = "color",
    type = "full", order = "hclust", p.mat = cormatrixall2$p,
    sig.level = 0.05, insig = "blank", tl.col = "black",
    outline = "white", title = "Correlation Matrix of Explanatory and Outcome
        Variables",
    line = -1, cex.main = 2, adj.main = 0.5)
```



9.3 STARD Flow Diagram

9.3.0.0.0.1 See instructions from https://www.r-bloggers.com/flow-charts-in-r/

9.3.0.0.0.2 Produces flow charts in Figure 1, (images need to be stretched out, output as svgs)

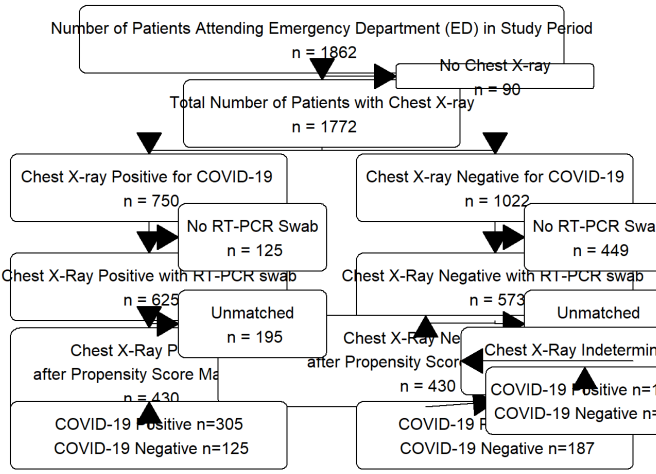
```
library(grid)
library(Gmisc)

grid.newpage()
# set some parameters to use repeatedly
leftx <- 0.25
```

9.3 STARD Flow Diagram 47

```
midx <- 0.5
rightx <- 0.75
width <- 0.4
gp <- gpar(fill = "white")
# create boxes
(totalattendance <- boxGrob("Number of Patients Attending Emergency Department
        (ED) in Study Period\n n = 1862",
    x = midx, y = 0.9, box_gp = gp, width = 0.7))

(numberwithxr <- boxGrob("Total Number of Patients with Chest X-ray\n n =
        1772",
    x = midx, y = 0.75, box_gp = gp, width = width))
# connect boxes like this
connectGrob(totalattendance, numberwithxr,
    "v")

(numberwithoutxr <- boxGrob("No Chest X-ray\n n = 90",
    x = rightx, y = 0.825, box_gp = gp, width = unit(2,
        "inch"), height = 0.05))

connectGrob(totalattendance, numberwithoutxr,
    "-")

(XRPos <- boxGrob("Chest X-ray Positive for COVID-19 \n n = 750",
    x = leftx, y = 0.6, box_gp = gp, width = width))
(XRNeg <- boxGrob("Chest X-ray Negative for COVID-19\n n = 1022",
    x = rightx, y = 0.6, box_gp = gp, width = width))

connectGrob(numberwithxr, XRPos, "N")
connectGrob(numberwithxr, XRNeg, "N")

(RTPCRXRPos <- boxGrob("Chest X-Ray Positive with RT-PCR swab\n n = 625",
    x = leftx, y = 0.4, box_gp = gp, width = width))
(RTPCRXRNeg <- boxGrob("Chest X-Ray Negative with RT-PCR swab \n n = 573",
    x = rightx, y = 0.4, box_gp = gp, width = width))

connectGrob(XRPos, RTPCRXRPos, "N")
connectGrob(XRNeg, RTPCRXRNeg, "N")

(NoRTPCRXRPos <- boxGrob("No RT-PCR Swab\n n = 125",
    x = 0.4, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))
(NoRTPCRXRNeg <- boxGrob("No RT-PCR Swab\n n = 449",
    x = 0.9, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))

connectGrob(XRPos, NoRTPCRXRPos, "-")
connectGrob(XRNeg, NoRTPCRXRNeg, "-")

(MatchedXRPos <- boxGrob("Chest X-Ray Positive \nafter Propensity Score
        Matching\n n = 430",
    x = leftx, y = 0.225, box_gp = gp, width = width))
(MatchedXRNeg <- boxGrob("Chest X-Ray Negative \nafter Propensity Score
        Matching \n n = 430",
    x = 0.65, y = 0.25, box_gp = gp, width = unit(4.2,
        "inch")))

connectGrob(RTPCRXRPos, MatchedXRPos, "N")
connectGrob(RTPCRXRNeg, MatchedXRNeg, "N")
```

```
(UnmatchedXRPos <- boxGrob("Unmatched\n n = 195",
    x = 0.4, y = 0.325, box_gp = gp, width = unit(1.5,
        "inch")))
(UnmatchedXRNeg <- boxGrob("Unmatched\n n = 143",
    x = 0.9, y = 0.325, box_gp = gp, width = unit(1.5,
        "inch")))

connectGrob(RTPCRXRPos, UnmatchedXRPos, "-")
connectGrob(RTPCRXRNeg, UnmatchedXRNeg, "L")

(DiagXRPositive <- boxGrob("COVID-19 Positive n=305\n COVID-19 Negative n=125",
    x = leftx, y = 0.1, box_gp = gp, width = width))
(DiagXRNegative <- boxGrob("COVID-19 Positive n=243 \n COVID-19 Negative
        n=187",
    x = rightx, y = 0.1, box_gp = gp, width = width))

connectGrob(MatchedXRPos, DiagXRPositive,
    "N")
connectGrob(MatchedXRNeg, DiagXRNegative,
    "vertical")


(XRInd <- boxGrob("Chest X-Ray Indeterminate \n n = 197",
    x = 0.88, y = 0.25, box_gp = gp, width = unit(2.5,
        "inch")))

connectGrob(MatchedXRNeg, XRInd, "horizontal")

(DiagXRInd <- boxGrob("COVID-19 Positive n=136\n COVID-19 Negative n=63",
    x = 0.88, y = 0.17, box_gp = gp, width = unit(2,
        "inch")))
connectGrob(XRInd, DiagXRInd, "vertical")
```

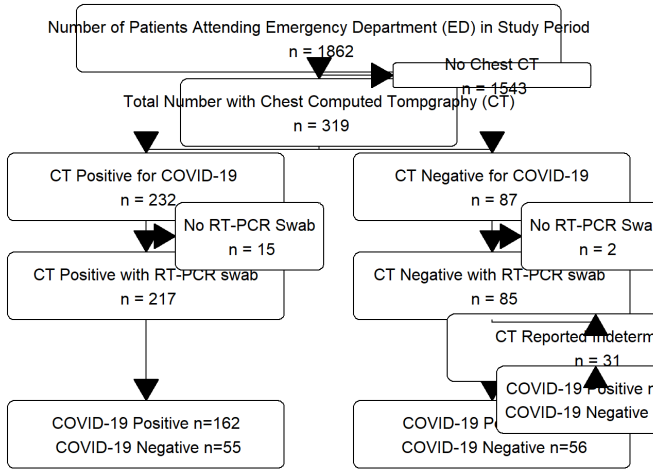
9.3 STARD Flow Diagram 49




```
##### CT FLow Chart#####

grid.newpage()
(totalattendance <- boxGrob("Number of Patients Attending Emergency Department
         (ED) in Study Period\n n = 1862",
    x = midx, y = 0.9, box_gp = gp, width = 0.7))

(numberwithCT <- boxGrob("Total Number with Chest Computed Tompgraphy (CT)\n n
        = 319",
    x = midx, y = 0.75, box_gp = gp, width = width))
connectGrob(totalattendance, numberwithCT,
    "vertical")

(numberwithoutCT <- boxGrob("No Chest CT\n n = 1543",
    x = rightx, y = 0.825, box_gp = gp, width = unit(2,
        "inch"), height = 0.05))

connectGrob(totalattendance, numberwithoutCT,
    "-")


(CTPos <- boxGrob("CT Positive for COVID-19 \n n = 232",
    x = leftx, y = 0.6, box_gp = gp, width = width))
(CTNeg <- boxGrob("CT Negative for COVID-19\n n = 87",
    x = rightx, y = 0.6, box_gp = gp, width = width))

connectGrob(numberwithCT, CTPos, "N")
connectGrob(numberwithCT, CTNeg, "N")

(RTPCRCTPos <- boxGrob("CT Positive with RT-PCR swab\n n = 217",
    x = leftx, y = 0.4, box_gp = gp, width = width))
```

```
(RTPCRCTNeg <- boxGrob("CT Negative with RT-PCR swab \n n = 85",
    x = rightx, y = 0.4, box_gp = gp, width = width))

connectGrob(CTPos, RTPCRCTPos, "N")
connectGrob(CTNeg, RTPCRCTNeg, "N")

(NoRTPCRCTPos <- boxGrob("No RT-PCR Swab\n n = 15",
    x = 0.4, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))
(NoRTPCRCTNeg <- boxGrob("No RT-PCR Swab\n n = 2",
    x = 0.9, y = 0.5, box_gp = gp, width = unit(1.5,
        "inch")))

connectGrob(CTPos, NoRTPCRCTPos, "-")
connectGrob(CTNeg, NoRTPCRCTNeg, "-")

(DiagCTPositive <- boxGrob("COVID-19 Positive n=162\n COVID-19 Negative n=55",
    x = leftx, y = 0.1, box_gp = gp, width = width))
(DiagCTNegative <- boxGrob("COVID-19 Positive n=29\n COVID-19 Negative n=56",
    x = rightx, y = 0.1, box_gp = gp, width = width))

connectGrob(RTPCRCTPos, DiagCTPositive, "N")
connectGrob(RTPCRCTNeg, DiagCTNegative, "N")


(CTInd <- boxGrob("CT Reported Indeterminate \n n = 31",
    x = 0.9, y = 0.275, box_gp = gp, width = unit(3,
        "inch")))

connectGrob(RTPCRCTNeg, CTInd, "N")

(DiagCTInd <- boxGrob("COVID-19 Positive n=16\n COVID-19 Negative n=15",
    x = 0.9, y = 0.17, box_gp = gp, width = unit(2,
        "inch")))
connectGrob(CTInd, DiagCTInd, "vertical")
```

9.3 STARD Flow Diagram 51




```
### Labels####
grid.newpage()
(indextest <- boxGrob("Index Tests", x = midx,
    y = 0.9, box_gp = gpar(fill = "light blue"),
    width = 0.7))

(reftest <- boxGrob("Index Tests and Reference Standards",
    x = midx, y = 0.4, box_gp = gpar(fill = "light blue"),
    width = 0.7))

(finaldiag <- boxGrob("Final Diagnoses",
    x = midx, y = 0.1, box_gp = gpar(fill = "light blue"),
    width = 0.7))
```

52 9 Pooled Regression after Multi…