Prevalence of Heart Disease in Framington, MA

Dustin Leatherman

2/15/2020

Introduction

Data Analysis

```
heart.clean <-
  heart %>%
  # there are several discrete features with low cardinality so
  # treating them as factors
  mutate(
    isMale = factor(male),
    education.factor = factor(education),
    currentSmoker.factor = factor(currentSmoker),
    prevalentStroke.factor = factor(prevalentStroke),
    prevalentHyp.factor = factor(prevalentHyp),
    diabetes.factor = factor(diabetes),
    TenYearCHD.factor = factor(TenYearCHD),
    BPMeds.factor = factor(BPMeds)
  ) %>%
  select(-c(prevalentStroke, male, education, prevalentHyp, diabetes, currentSmoker, TenYearCHD, BPMeds

heart.clean %>% ggpairs(
  aes(color = TenYearCHD.factor, alpha = 0.3),
    # correlation text is off so this makes it readable
   upper = list(continuous = wrap("cor", size = 3, hjust=0, alignPercent=1)),
   title = c("Scatterplot by Ten Year CHD"),
  ) + labs(caption = "Figure 1. Relationships between features are explored in regards to whether or no
```

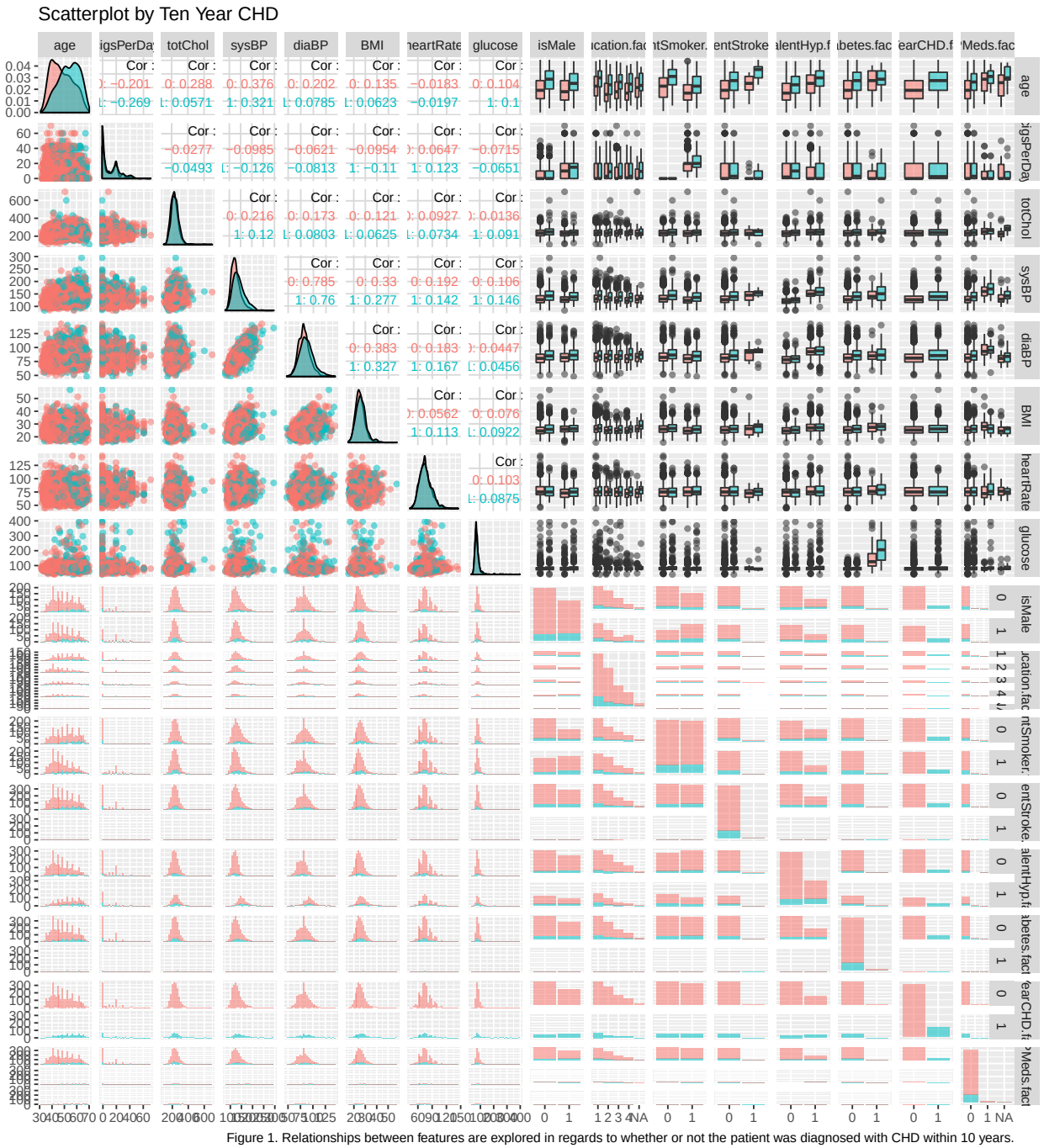


Figure 1. Relationships between features are explored in regards to whether or not the patient was diagnosed with CHD within 10 years.

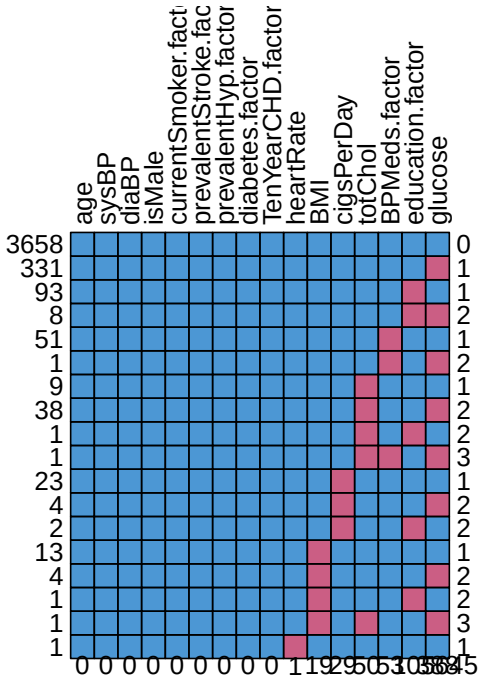
Observations

- The Age density chart indicates that a larger number of older patients were diagnosed than younger patients. The age in which both meet is around 50.
- Cigs Per Day, total Cholesterol, Systolic BP, Diastolic BP, BMI, Heart Rate, and Glucose levels *all* appear to be right-skewed according to their Density Charts. Typically a log transformation is applied in such cases, but may not be the best choice since we will be applying Logistic Regression later on.
- The only strong correlation present is between Diastolic BP and Systolic BP. These are similar for patients with and without CDH. Likely one of these variables can be discarded.
- The scatterplots comparing Glucose vs Systolic BP, Diastolic BP, and BMI show that there appear to be higher glucose levels for those who have been diagnosed with CHD.

- A similar pattern is seen in the Diabetes vs Glucose box plots. There is a wider range of Glucose levels between the 25th and 75th quantiles and a significantly higher median for those diagnosed with CDH. This indicates glucose levels may be a relevant predictor.
- The prevalent Hypertension vs Age Boxplot indicates that the 25th, 50yh, and 75th quantile values are larger for older patients with prevalent Hypertension. The values are even larger for those who have been diagnosed with CHD indicating Prevalent Hypertension may be associated with CHD.
- The Prevalent Stroke vs Age Boxplot shows similar characteristics to the Prevalent Hypertension vs Age Boxplot indicating that there may be a relationship with CHD.

Missing Data

```
# show missingness Graph
md.pattern(heart.clean, rotate.names = TRUE)
```



There are 645 rows which contain missing data. The indicator graph shows that missing data typically falls into a select few fields. This indicates that the data is not Missing Completely at Random (MCAR).

```
# Run Little's Test to determine if the data is Missing Completely at Random (MCAR)
LittleMCAR(heart.clean)$p.value
```

```
## this could take a while
```

```
## [1] 4.955219e-08
```

There is convincing evidence that the missing data is not completely random (Little's Test). Thus, it is inappropriate to drop the data as it would be dropping meaningful patterns from the analysis. Therefore, the missing values will be imputed using Multiple Imputation with Markov Chain Monte Carlo simulations.

Analysis

```
# create a 30% sample for training data. The 30% is arbitrary
heart.samples <-
  stratified(heart.clean, c("TenYearCHD.factor"), .3, bothSets = TRUE)

heart.testing <- heart.samples$SAMP1  %>%
  mutate(TenYearCHD = as.numeric(as.character(TenYearCHD.factor))) %>%
    select(-education.factor)

heart.training <- heart.samples$SAMP2  %>%
  mutate(TenYearCHD = as.numeric(as.character(TenYearCHD.factor))) %>%
    select(-education.factor)

heart.impute.testing <- mice(heart.testing, m = 10, maxit = 50, seed = 123)
heart.impute.testing.complete <- mice::complete(heart.impute.testing)

heart.impute.training <- mice(heart.training, m = 10, maxit = 50, seed = 123)
heart.impute.training.complete <- mice::complete(heart.impute.training)
```

```
fit.reg <- glm(
    TenYearCHD ~ age + sysBP + BMI + glucose +  prevalentHyp.factor + prevalentStroke.factor + BPMeds.fa
    family = "binomial",
    data = heart.impute.training.complete
  )

tidy(fit.reg) %>%
  kable(
    digits = 4
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

term	estimate	std.error	statistic	p.value
(Intercept)	-8.1611	0.7160	-11.3984	0.0000
age	0.0623	0.0072	8.6333	0.0000
sysBP	0.0115	0.0033	3.4885	0.0005
BMI	-0.0010	0.0135	-0.0755	0.9398
glucose	0.0084	0.0020	4.1315	0.0000
prevalentHyp.factor1	0.3155	0.1544	2.0431	0.0410
prevalentStroke.factor1	0.8289	0.4990	1.6611	0.0967
BPMeds.factor1	0.2510	0.2528	0.9928	0.3208
totChol	0.0021	0.0012	1.8076	0.0707
cigsPerDay	0.0215	0.0047	4.6096	0.0000
heartRate	-0.0012	0.0047	-0.2529	0.8004
isMale1	0.4503	0.1193	3.7738	0.0002

There are a handful of variables that are considered not significant in predicting risk for CHD.

```
car::vif(fit.reg) %>% kable(
  caption = "Variance Inflation Factors for CHD Predictors"
) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

The low Variance Inflation Factors indicate that multicollinearity between predictors is not significantly present. This indicates that the high p-values are likely related to being statistically insignificant opposed to

Table 1: Variance Inflation Factors for CHD Predictors

	x
age	1.221065
sysBP	2.227995
BMI	1.123175
glucose	1.026837
prevalentHyp.factor	2.036167
prevalentStroke.factor	1.025618
BPMeds.factor	1.107784
totChol	1.049018
cigsPerDay	1.262697
heartRate	1.100576
isMale	1.220023

its information already being included in the model via other predictors.

```
fit.reg.reduced <- glm(
    TenYearCHD ~ age + sysBP + glucose + cigsPerDay + isMale + totChol,
    family = "binomial",
    data = heart.impute.training.complete
  )

lmtest::lrtest(fit.reg, fit.reg.reduced) %>%
  tidy %>%
  kable(
    digits = 4,
    caption = "Likelihood Ratio Test. Comparing Full vs Reduced Model"
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

Table 2: Likelihood Ratio Test. Comparing Full vs Reduced Model

X.Df	LogLik	df	statistic	p.value
12	-1126.171	NA	NA	NA
7	-1130.633	-5	8.9245	0.1121

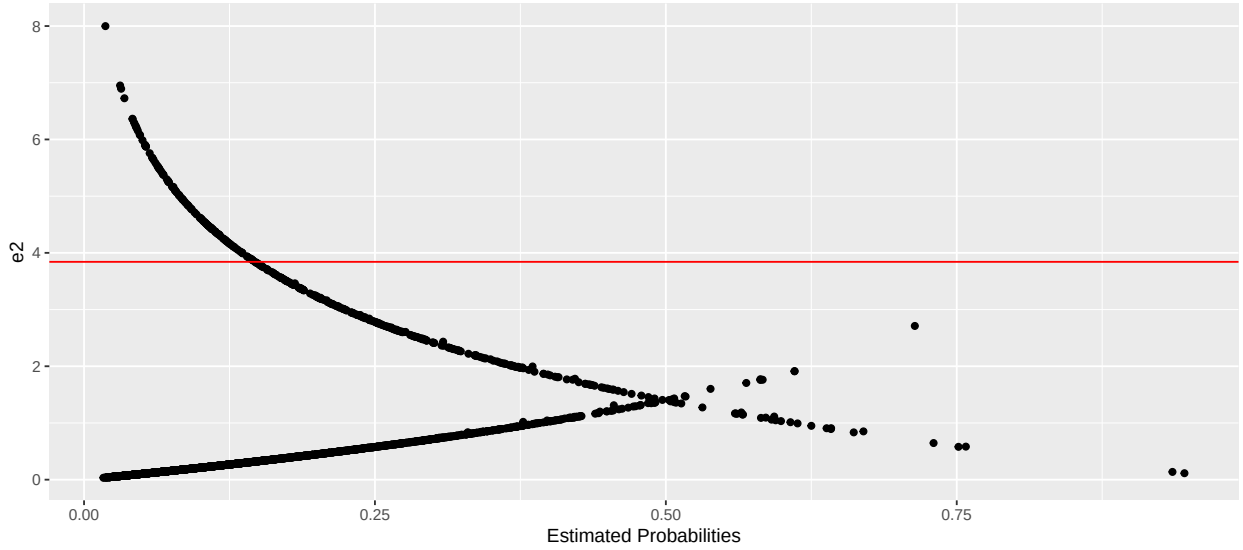
Removing the insignificant parameters from the full model and comparing with the significant values in a reduced model shows that there is no evidence that the full model explains more deviance than the reduced model. Further model comparisons were conducted on interaction and quadratic terms with no differing results. Going forward, this reduced model is what will be used.

```
heart.evp <-
  heart.impute.training.complete %>%
  select(TenYearCHD, age, sysBP, glucose, cigsPerDay, isMale, totChol) %>%
  group_by(age, sysBP, glucose, cigsPerDay, isMale, totChol) %>%
  summarize_all(
    funs(
      n = n(),
      y = sum(TenYearCHD),
      fail = n - y
    )
  ) %>% ungroup
```

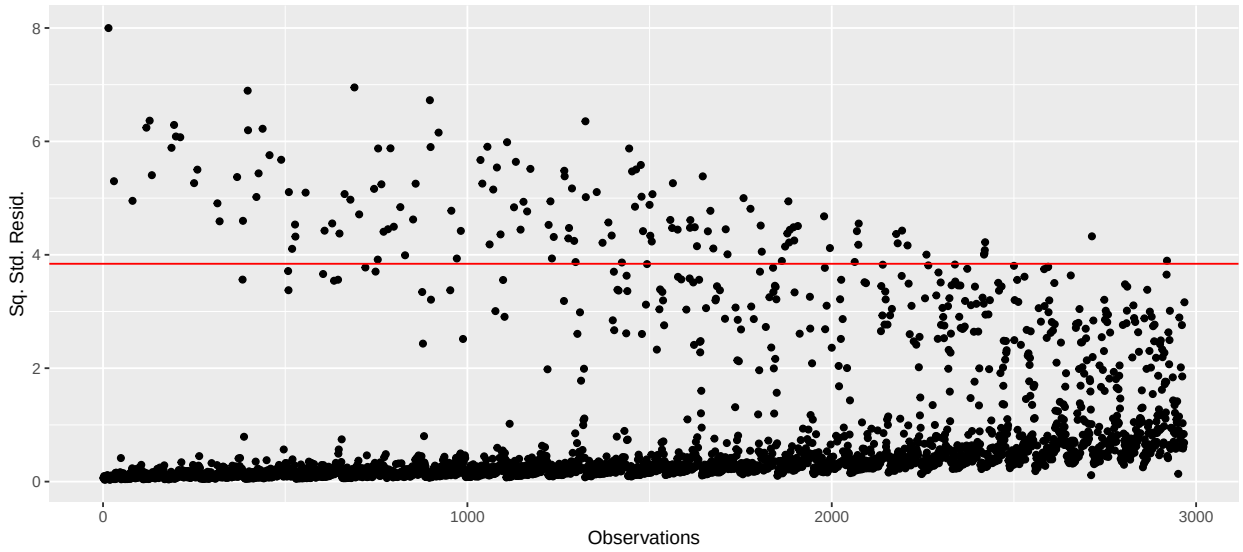
```
fit.reg.reduced.evp <- glm(
    y/n ~ age + sysBP + glucose + cigsPerDay + isMale + totChol,
    family = "binomial",
    data = heart.evp,
    weights = n
  )

values <- augment(fit.reg.reduced.evp, type.residuals = "pearson") %>%
  mutate(
    e2 = .std.resid^2,
    p = exp(.fitted)/(1 + exp(.fitted))
  )

values %>% head %>% kable(digits = 4)
```

y.n	age	sysBP	glucose	cigsPerDay	isMale	totChol	X.weights.	.fitted	.se.fit	.resid	.hat	.sigma
0	32	111.0	88	15	0	242	1	-3.4477	0.1687	-0.1784	0.0009	0.874
0	33	116.0	93	15	0	199	1	-3.3554	0.1685	-0.1868	0.0009	0.874
0	33	141.5	77	0	1	165	1	-2.9974	0.1998	-0.2234	0.0018	0.874
0	34	92.5	68	10	0	159	1	-4.0791	0.1955	-0.1301	0.0006	0.874
0	34	100.5	115	5	1	185	1	-3.1454	0.1870	-0.2075	0.0014	0.874
0	34	102.0	75	0	0	227	1	-3.9218	0.1752	-0.1407	0.0006	0.874

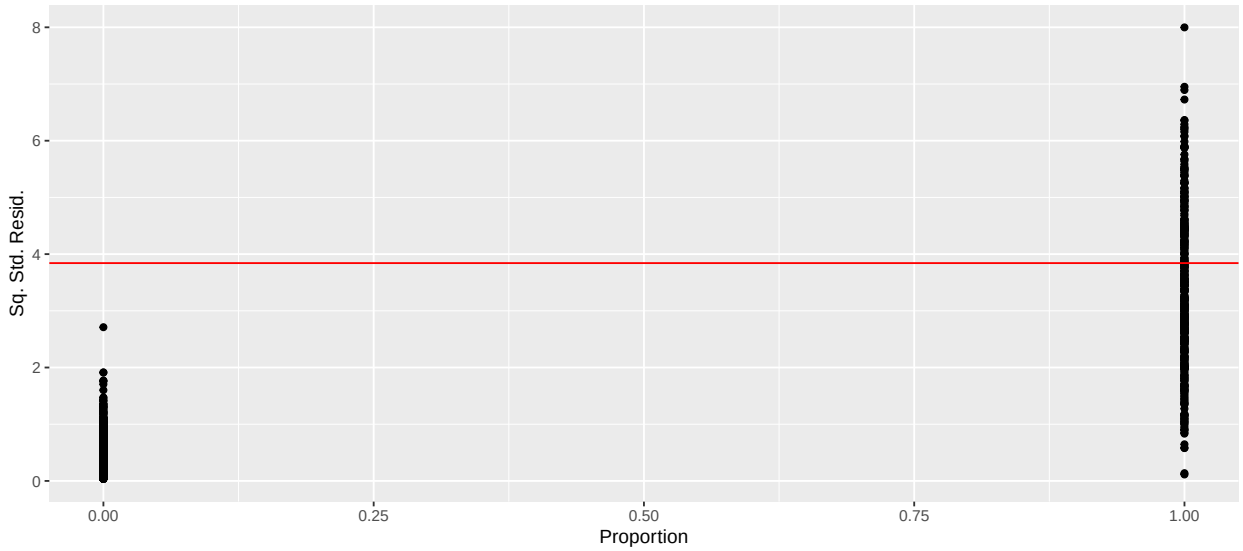
```
qplot(data = values, p, e2, xlab = "Estimated Probabilities") +
    geom_hline(yintercept = qchisq(0.95, 1), color = "red")
```




```
qplot(data = values, 1:nrow(values), e2, xlab = "Observations", ylab = "Sq. Std. Resid.") +
    geom_hline(yintercept = qchisq(0.95, 1), color = "red")
```

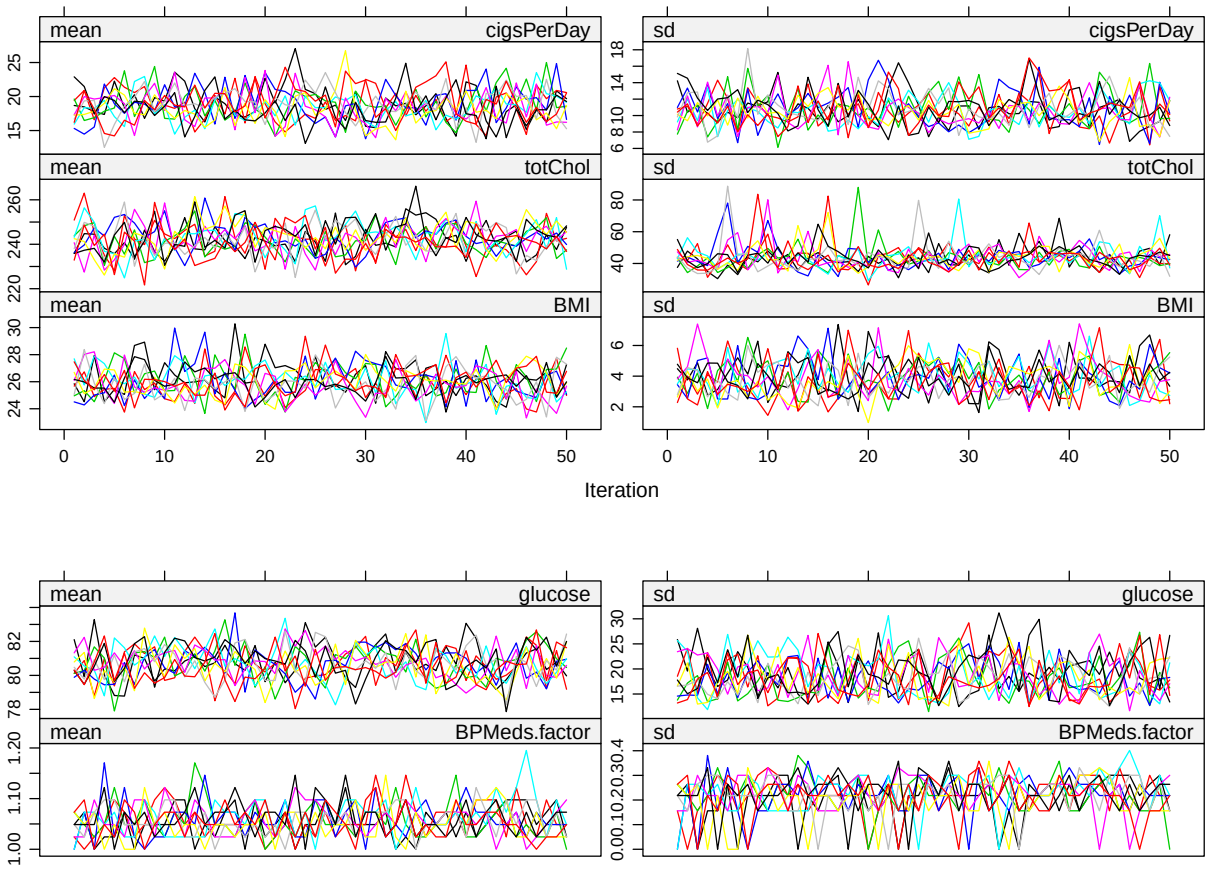



```
qplot(data = values, y.n, e2, xlab = "Proportion", ylab = "Sq. Std. Resid.") +
    geom_hline(yintercept = qchisq(0.95, 1), color = "red")
```



Imputed Data

```
plot(heart.impute.training)
```



Over 50 iterations of imputed values, it is ideal to see that the lines in both the mean and standard deviation intermingle and be free of any trends as the number of iterations increase. A seed is used in order to provide reproducability in the generation of values. The lines appear to intermingle and no significant trends are visible.

```
# fit our logistic regression model on the imputed values
fit.imp <- with(data = heart.impute.training, glm(TenYearCHD ~ age + sysBP + glucose + cigsPerDay + isM

# pool the imputations together
fit.pool <- mice::pool(fit.imp)

fit.pool$pooled %>% as_tibble(rownames = "term") %>%
  kable(
    digits = 4
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

term	estimate	ubar	b	t	dfcom	df	riv	lambda	fmi
(Intercept)	-8.8240	0.2716	0.0024	0.2743	2961	2843.168	0.0098	0.0097	0.0104
age	0.0639	0.0001	0.0000	0.0001	2961	2953.746	0.0013	0.0013	0.0019
sysBP	0.0164	0.0000	0.0000	0.0000	2961	2947.597	0.0022	0.0022	0.0029
glucose	0.0083	0.0000	0.0000	0.0000	2961	2150.062	0.0334	0.0323	0.0332
cigsPerDay	0.0207	0.0000	0.0000	0.0000	2961	2954.854	0.0010	0.0010	0.0017
isMale1	0.4547	0.0138	0.0000	0.0138	2961	2954.707	0.0011	0.0011	0.0017
totChol	0.0019	0.0000	0.0000	0.0000	2961	2217.672	0.0313	0.0304	0.0312

The fractional information missing due to nonresponse (fmi) and the relative in crease in variance due to nonresponse are low which indicates the imputed data doesn't have a significant effect on the shape of the data itself.

```
summary(fit.pool) %>% as_tibble(rownames = "term") %>%
  kable (
    digits = 4
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

term	estimate	std.error	statistic	df	p.value
(Intercept)	-8.8240	0.5237	-16.8490	2843.168	0.0000
age	0.0639	0.0071	8.9385	2953.746	0.0000
sysBP	0.0164	0.0024	6.9352	2947.597	0.0000
glucose	0.0083	0.0020	4.1214	2150.062	0.0000
cigsPerDay	0.0207	0.0046	4.4965	2954.854	0.0000
isMale1	0.4547	0.1174	3.8748	2954.707	0.0001
totChol	0.0019	0.0012	1.5822	2217.672	0.1137

Results

Single Data Imputed Model

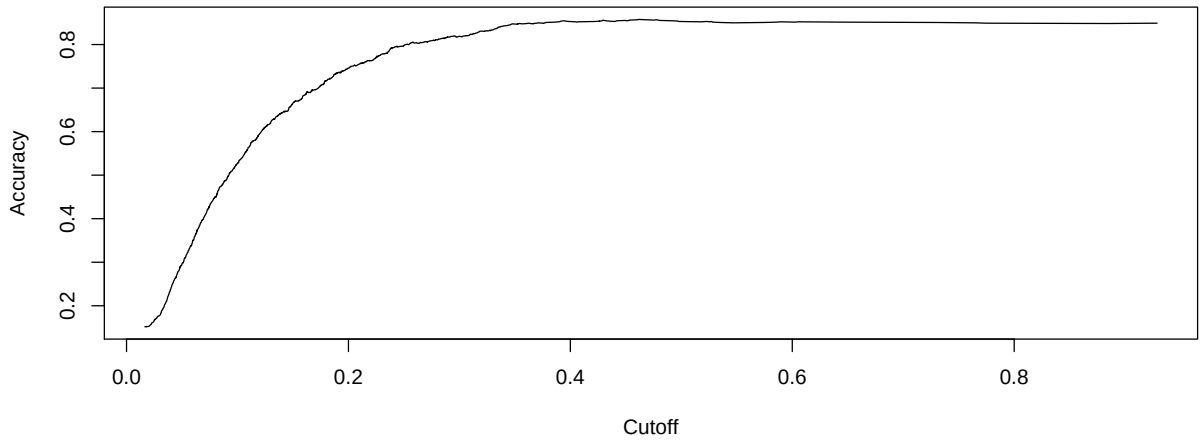
```
pred <- predict(fit.reg.reduced, newdata = heart.impute.testing.complete, type = "response")
confusionMatrix(table(as.numeric(pred > 0.5), heart.testing %>% select(TenYearCHD) %>% as_vector()))
```

```
## Confusion Matrix and Statistics
##
##
##        0    1
##   0 1072  181
##   1    7   12
##
##                Accuracy : 0.8522
##                  95% CI : (0.8315, 0.8713)
##     No Information Rate : 0.8483
##     P-Value [Acc > NIR] : 0.3655
##
##                   Kappa : 0.0884
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.99351
##             Specificity : 0.06218
```

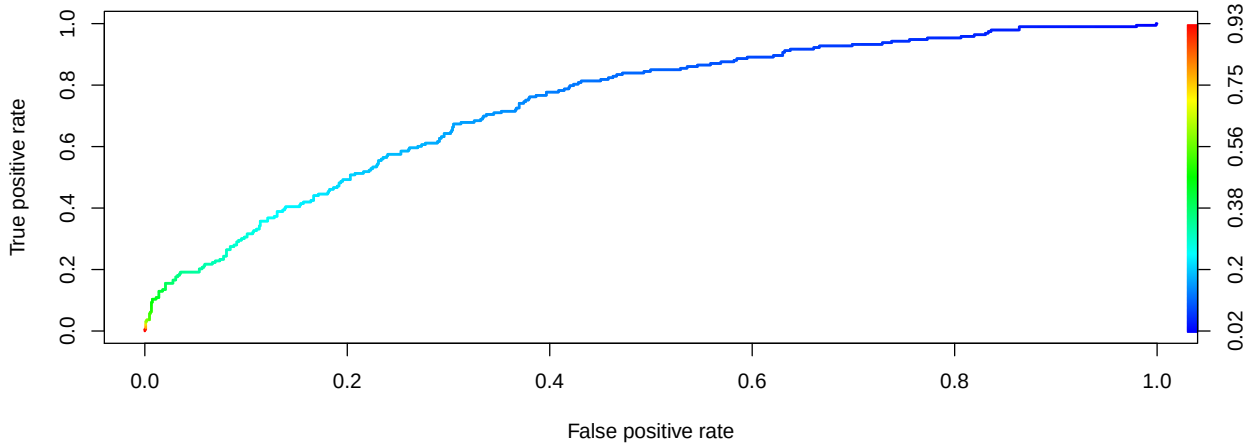
```
##          Pos Pred Value : 0.85555
##          Neg Pred Value : 0.63158
##              Prevalence : 0.84827
##          Detection Rate : 0.84277
##    Detection Prevalence : 0.98506
##       Balanced Accuracy : 0.52784
##
##        'Positive' Class : 0
##
```

The Predictive power is approximately 85%, with the dropped data being slightly better than the imputed data model. The above matrix uses 0.5 as the cutoff threshold. Due to the gravity of a False Negative for detecting disease, the threshold can be adjusted as needed.

```
pred2 <- prediction(pred, heart.testing$TenYearCHD)
perf <- performance(pred2, "acc")
plot(perf, title = "ROC Curve")
```




```
perf2 <- performance(pred2, "tpr", "fpr")
plot(perf2, colorize = T, lwd = 2)
```



For the model against dropped data, the highest accuracy occurs levels off with a cutoff greater than 0.5. The ideal AUC curve would be in the upper left area of the graph. The initial take on using 0.5 as the cutoff is likely the correct choice

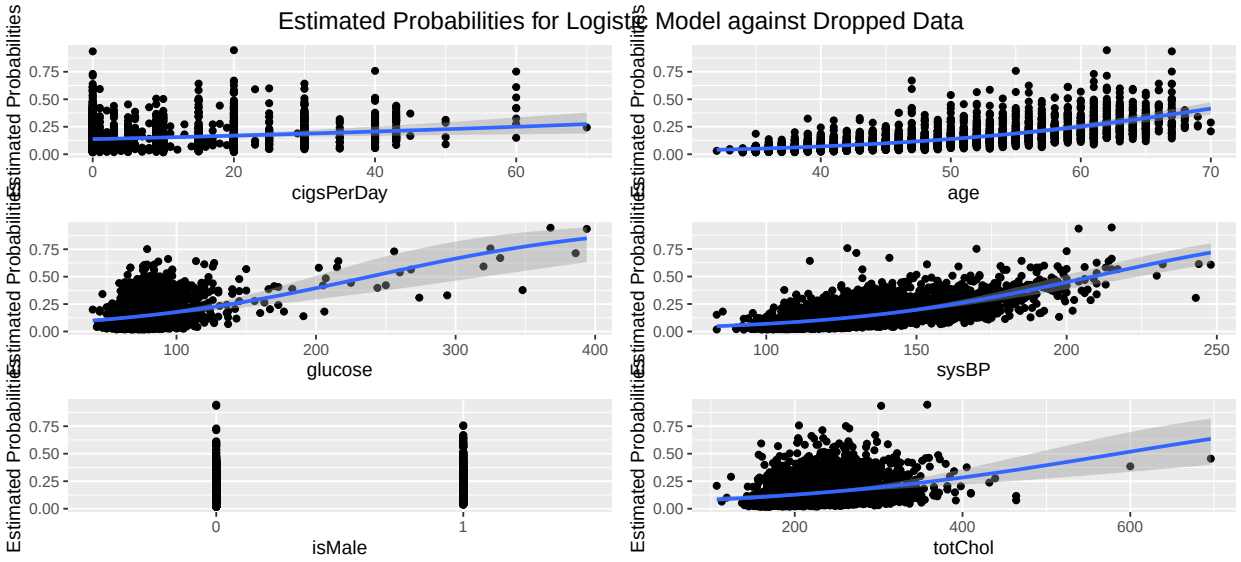
```
auc <- performance(pred2, "auc")
print(auc@y.values[[1]])
```

```
## [1] 0.7389734
```

An AUC as close to 1 is preferred. The above value is not bad but could be better.

```
plot.fit.reg.age <-
  fit.reg.reduced %>%
    ggplot(aes(x = age, y = exp(.fitted)/(1 + exp(.fitted)))) +
      geom_point() +
      geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.fit.reg.glucose <-
  fit.reg.reduced %>%
    ggplot(aes(x = glucose, y = exp(.fitted)/(1 + exp(.fitted)))) +
      geom_point() +
      geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.fit.reg.sysBP <-
  fit.reg.reduced %>%
    ggplot(aes(x = sysBP, y = exp(.fitted)/(1 + exp(.fitted)))) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.fit.reg.cigsPerDay <-
  fit.reg.reduced %>%
    ggplot(aes(x = cigsPerDay, y = exp(.fitted)/(1 + exp(.fitted)))) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
```

```
      ylab("Estimated Probabilities")

plot.fit.reg.isMale <-
  fit.reg.reduced %>%
    ggplot(aes(x = isMale, y = exp(.fitted)/(1 + exp(.fitted)))) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.fit.reg.totChol <-
  fit.reg.reduced %>%
    ggplot(aes(x = totChol, y = exp(.fitted)/(1 + exp(.fitted)))) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

grid.arrange(
  plot.fit.reg.cigsPerDay,
  plot.fit.reg.age,
  plot.fit.reg.glucose,
  plot.fit.reg.sysBP,
  plot.fit.reg.isMale,
  plot.fit.reg.totChol,
  ncol = 2,
  top = textGrob("Estimated Probabilities for Logistic Model against Dropped Data",
          gp=gpar(fontsize=14,font=1),just=c("center"))
  )
```



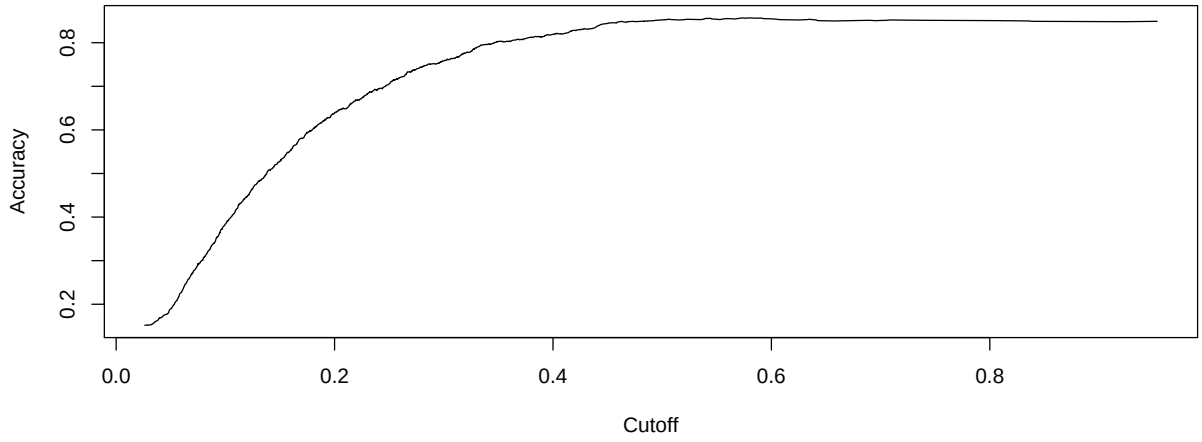
The strongest predictors for determining whether an individual is at risk for CHD are Systolic BP and Glucose. Initial analysis of the data showed Age being an obvious factor but the model does not seem to agree.

Imputed Pooled Data Model

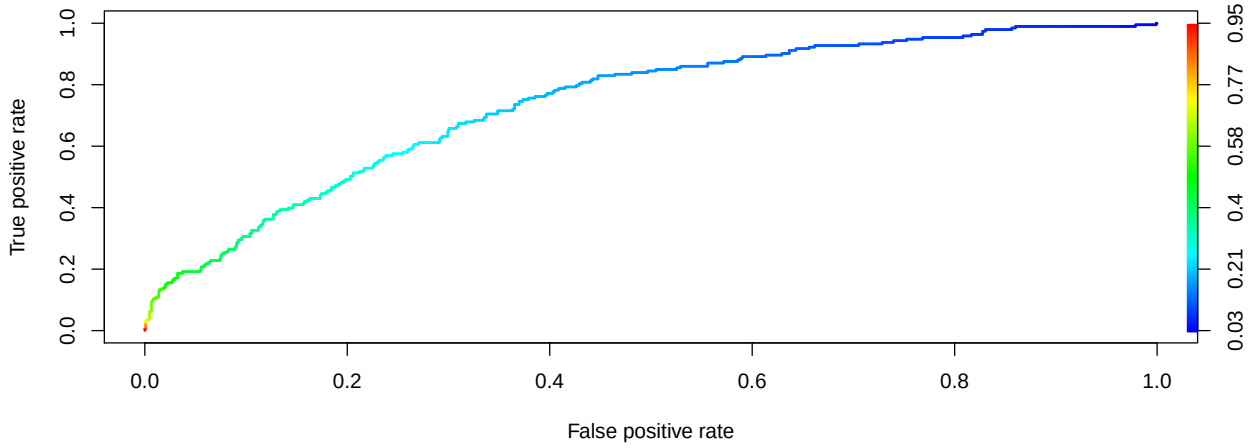
```r
# A good way doesnt exist to use predict() with a pooled model from mice.
# this predicts by doing the logistic calculation
dataToPredict <- heart.impute.testing.complete %>%
  mutate(intercept = 1) %>%
  select(intercept, age, sysBP, glucose, cigsPerDay, isMale, totChol)

# get our predicted estimates
nu <- summary(fit.pool)$estimate

# calculate the preducted values
predVals <- as.matrix(sapply(dataToPredict, as.numeric)) %*% nu

# calculate probabilites
confusion.pool <-
  predVals %>%
    as_tibble() %>%
    bind_cols(dataToPredict) %>%
    mutate(p = exp(V1) / (1 + exp(V1)))

confusionMatrix(
  table(
    as.numeric(confusion.pool %>% select(p) > 0.6),
    mice::complete(heart.impute.testing)
    %>% select(TenYearCHD) %>% as_vector()
  )
)
```

```
## Confusion Matrix and Statistics
##
##
##        0    1
##   0 1072  178
##   1    7   15
##
##                Accuracy : 0.8546
##                  95% CI : (0.834, 0.8735)
##     No Information Rate : 0.8483
##     P-Value [Acc > NIR] : 0.2809
##
##                   Kappa : 0.112
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.99351
##             Specificity : 0.07772
##          Pos Pred Value : 0.85760
##          Neg Pred Value : 0.68182
##              Prevalence : 0.84827
##          Detection Rate : 0.84277
##    Detection Prevalence : 0.98270
##       Balanced Accuracy : 0.53562
##
##        'Positive' Class : 0
##
```

```
pred2 <- prediction(confusion.pool %>% select(p), heart.testing$TenYearCHD)
perf <- performance(pred2, "acc")
plot(perf, title = "ROC Curve")
```




```
perf2 <- performance(pred2, "tpr", "fpr")
plot(perf2, colorize = T, lwd = 2)
```



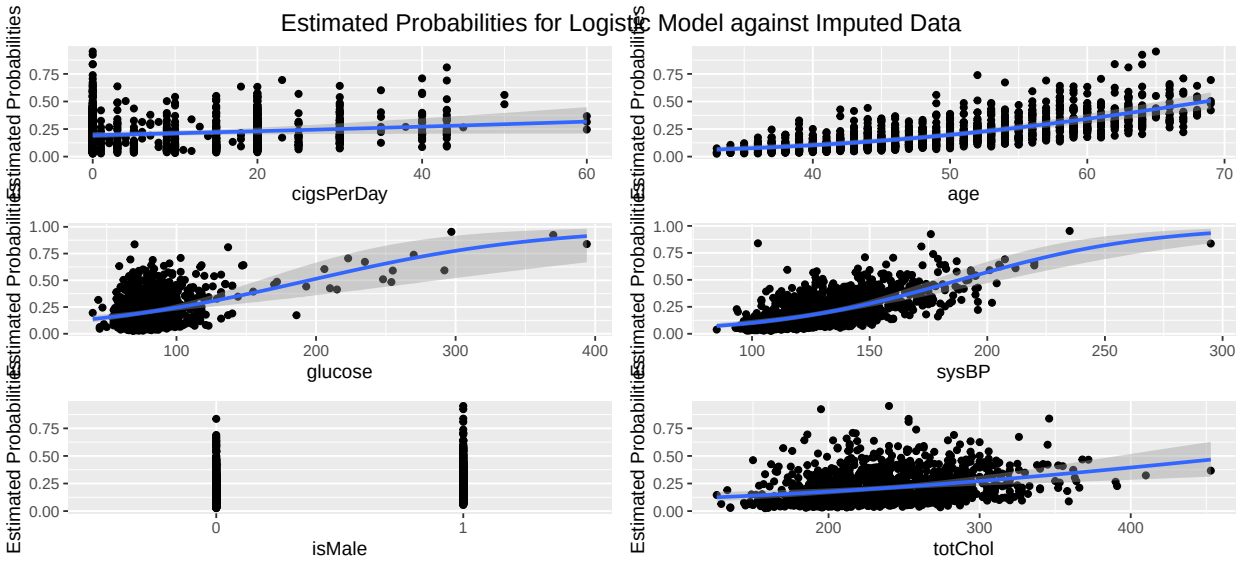

```
auc <- performance(pred2, "auc")
print(auc@y.values[[1]])
```

```
## [1] 0.7389878
```

The confusion matrix for the imputed data is similar to the dropped data model, as well as the Performance and AUC plots.

```
plot.pool.age <-
  confusion.pool %>%
    ggplot(aes(x = age, y = p)) +
      geom_point() +
```

```
      geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.pool.glucose <-
   confusion.pool %>%
    ggplot(aes(x = glucose, y = p)) +
      geom_point() +
      geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.pool.sysBP <-
  confusion.pool %>%
    ggplot(aes(x = sysBP, y =p )) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.pool.cigsPerDay <-
  confusion.pool %>%
    ggplot(aes(x = cigsPerDay, y = p)) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.pool.isMale <-
  confusion.pool %>%
    ggplot(aes(x = isMale, y = p)) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

plot.pool.totChol <-
  confusion.pool %>%
    ggplot(aes(x = totChol, y =p)) +
    geom_point() +
    geom_smooth(method = glm, method.args = list(family = "binomial")) +
      ylab("Estimated Probabilities")

grid.arrange(
  plot.pool.cigsPerDay,
  plot.pool.age,
  plot.pool.glucose,
  plot.pool.sysBP,
  plot.pool.isMale,
  plot.pool.totChol,
  ncol = 2,
  top = textGrob("Estimated Probabilities for Logistic Model against Imputed Data",
          gp=gpar(fontsize=14,font=1),just=c("center"))
  )
```



Again, similar behavior though less extreme is found with this one.

Conclusion

The pooled model against the imputed data proves to be similar as the one against the a single set of imputed data.