Predicting Diabetes in Pima Indians

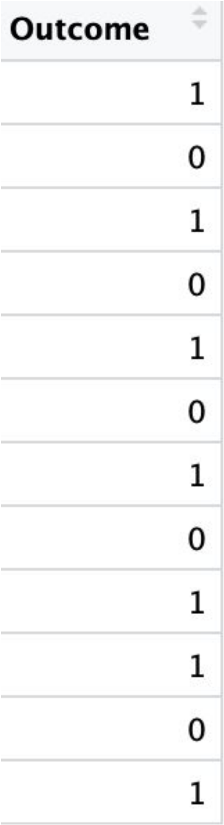
Nithu Mathew

Variables

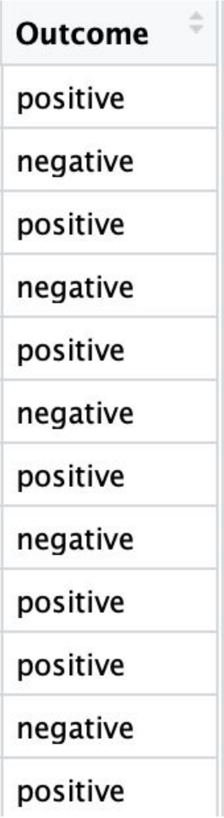
- The dataset consists of 768 instances and 9 attributes, one of which is a target variable (Outcome) and the other 8 being predictor variables.
- Attributes:
  1. Pregnancies (Number of times pregnant)
  2. Glucose (Plasma glucose concentration after 2 hours in an oral glucose tolerance test)
  3. BloodPressure (Diastolic blood pressure (mm Hg))
  4. SkinThickness (Triceps skin fold thickness (mm))
  5. Insulin (2-Hour serum insulin (mu U/ml))
  6. BMI (Body mass index (weight in kg/(height in m)^2))
  7. DiabetesPedigreeFunction
  8. Age
  9. Outcome

Data Cleaning

- Original dataset used binary numbers to indicate whether or not the subject had diabetes
- Factor level labels were changed to "positive" and "negative" to make it easier to identify.

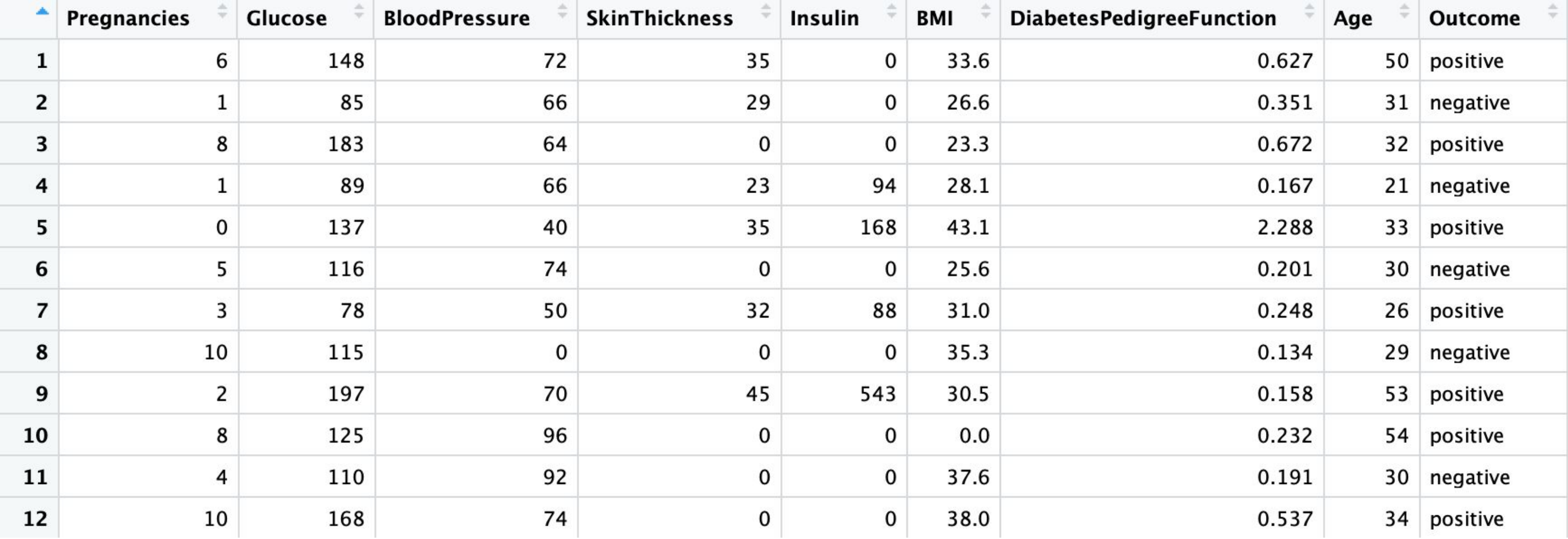


Outcome
1
0
1
0
1
0
1
0
1
1
0
1



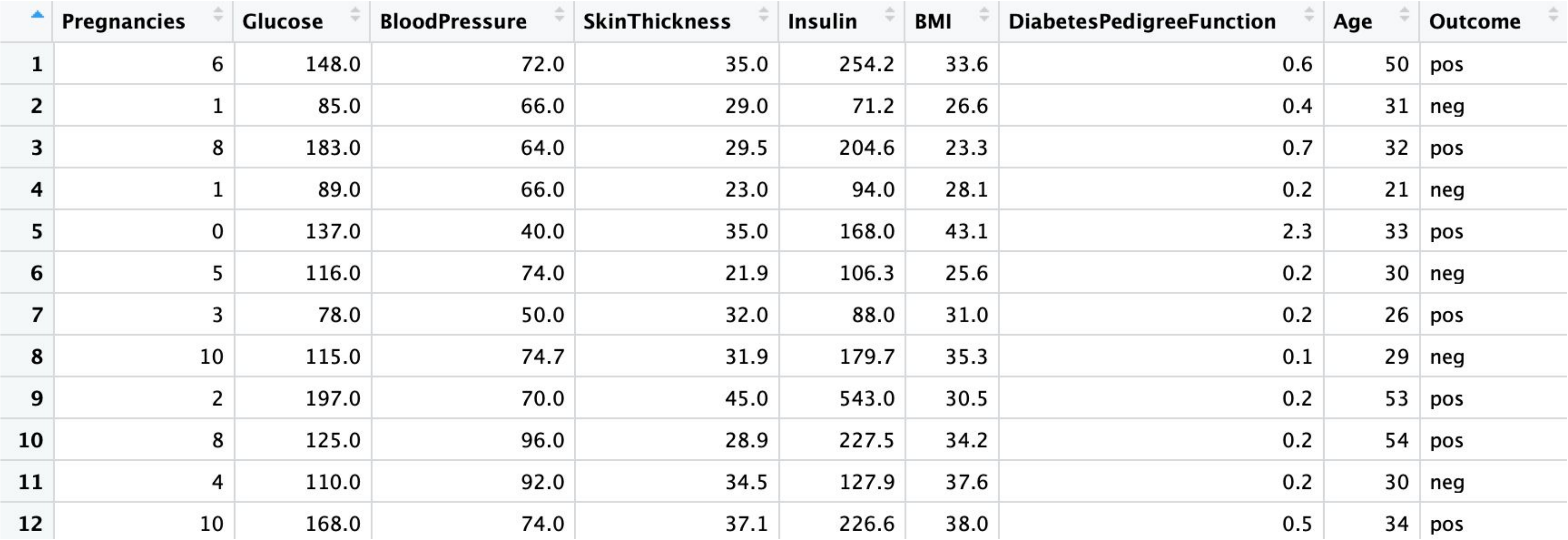
Outcome
positive
negative
positive
negative
positive
negative
positive
negative
positive
positive
negative
positive

Data Cleaning



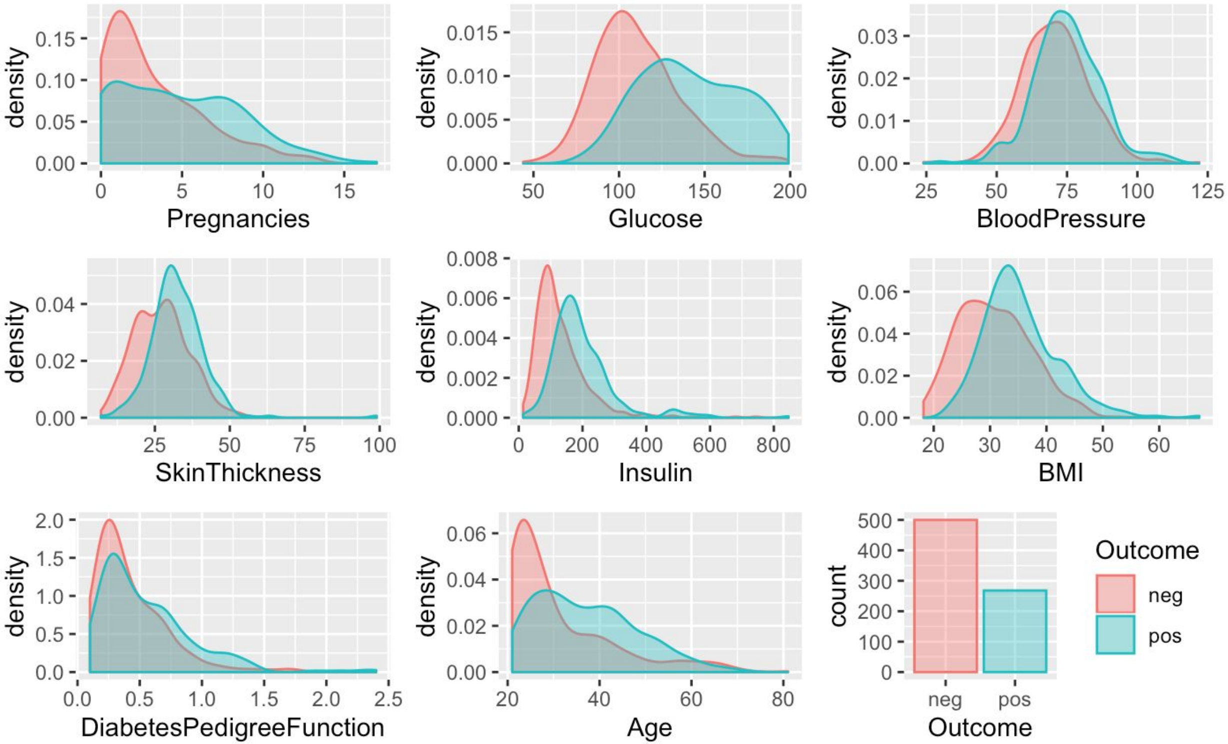
	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
1	6	148	72	35	0	33.6	0.627	50	positive
2	1	85	66	29	0	26.6	0.351	31	negative
3	8	183	64	0	0	23.3	0.672	32	positive
4	1	89	66	23	94	28.1	0.167	21	negative
5	0	137	40	35	168	43.1	2.288	33	positive
6	5	116	74	0	0	25.6	0.201	30	negative
7	3	78	50	32	88	31.0	0.248	26	positive
8	10	115	0	0	0	35.3	0.134	29	negative
9	2	197	70	45	543	30.5	0.158	53	positive
10	8	125	96	0	0	0.0	0.232	54	positive
11	4	110	92	0	0	37.6	0.191	30	negative
12	10	168	74	0	0	38.0	0.537	34	positive

K-Nearest Neighbour Imputation

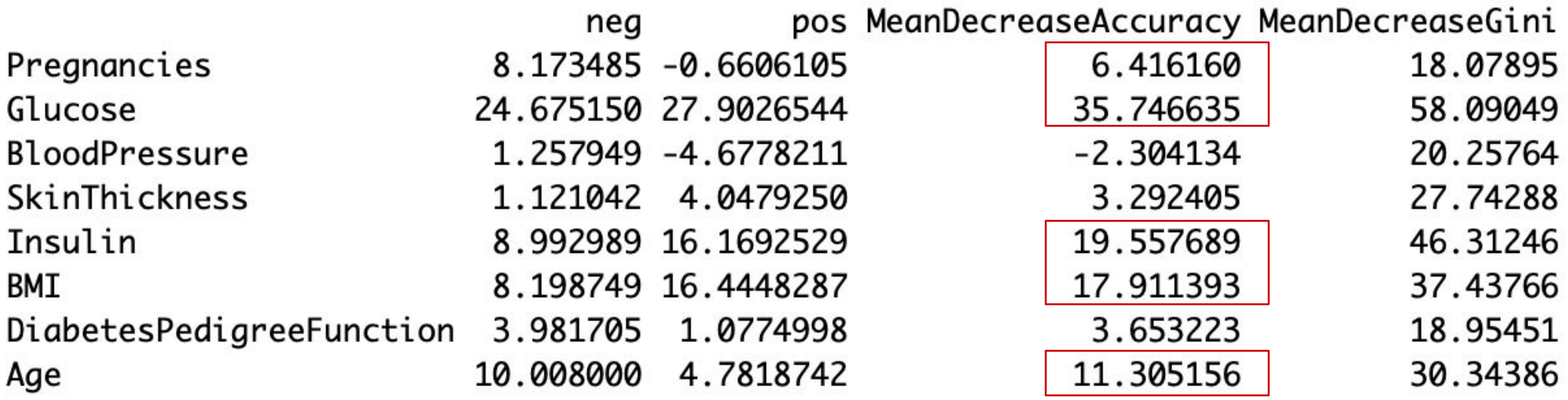


	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
1	6	148.0	72.0	35.0	254.2	33.6	0.6	50	pos
2	1	85.0	66.0	29.0	71.2	26.6	0.4	31	neg
3	8	183.0	64.0	29.5	204.6	23.3	0.7	32	pos
4	1	89.0	66.0	23.0	94.0	28.1	0.2	21	neg
5	0	137.0	40.0	35.0	168.0	43.1	2.3	33	pos
6	5	116.0	74.0	21.9	106.3	25.6	0.2	30	neg
7	3	78.0	50.0	32.0	88.0	31.0	0.2	26	pos
8	10	115.0	74.7	31.9	179.7	35.3	0.1	29	neg
9	2	197.0	70.0	45.0	543.0	30.5	0.2	53	pos
10	8	125.0	96.0	28.9	227.5	34.2	0.2	54	pos
11	4	110.0	92.0	34.5	127.9	37.6	0.2	30	neg
12	10	168.0	74.0	37.1	226.6	38.0	0.5	34	pos

Variable Distribution

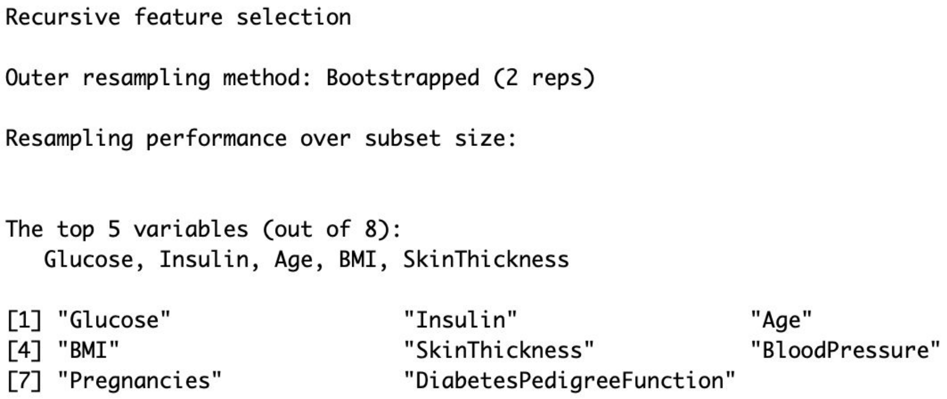


Feature Selection: Random Forests

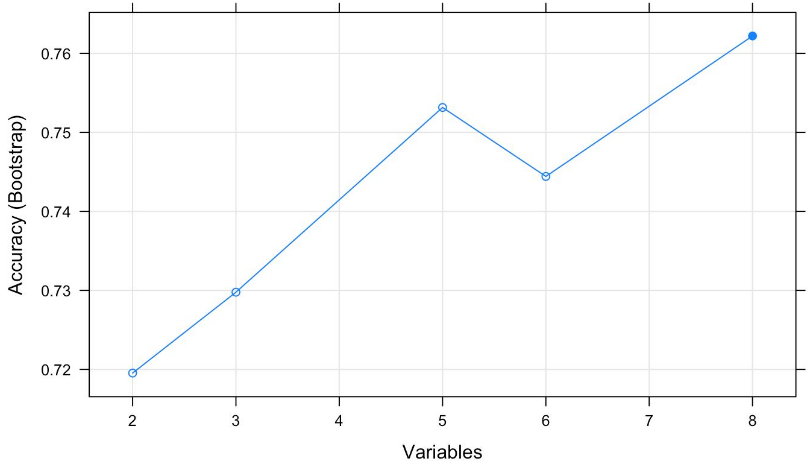


```
                               neg        pos MeanDecreaseAccuracy MeanDecreaseGini
Pregnancies               8.173485 -0.6606105             6.416160         18.07895
Glucose                  24.675150 27.9026544            35.746635         58.09049
BloodPressure             1.257949 -4.6778211            -2.304134         20.25764
SkinThickness             1.121042  4.0479250             3.292405         27.74288
Insulin                   8.992989 16.1692529            19.557689         46.31246
BMI                       8.198749 16.4448287            17.911393         37.43766
DiabetesPedigreeFunction  3.981705  1.0774998             3.653223         18.95451
Age                      10.008000  4.7818742            11.305156         30.34386
```

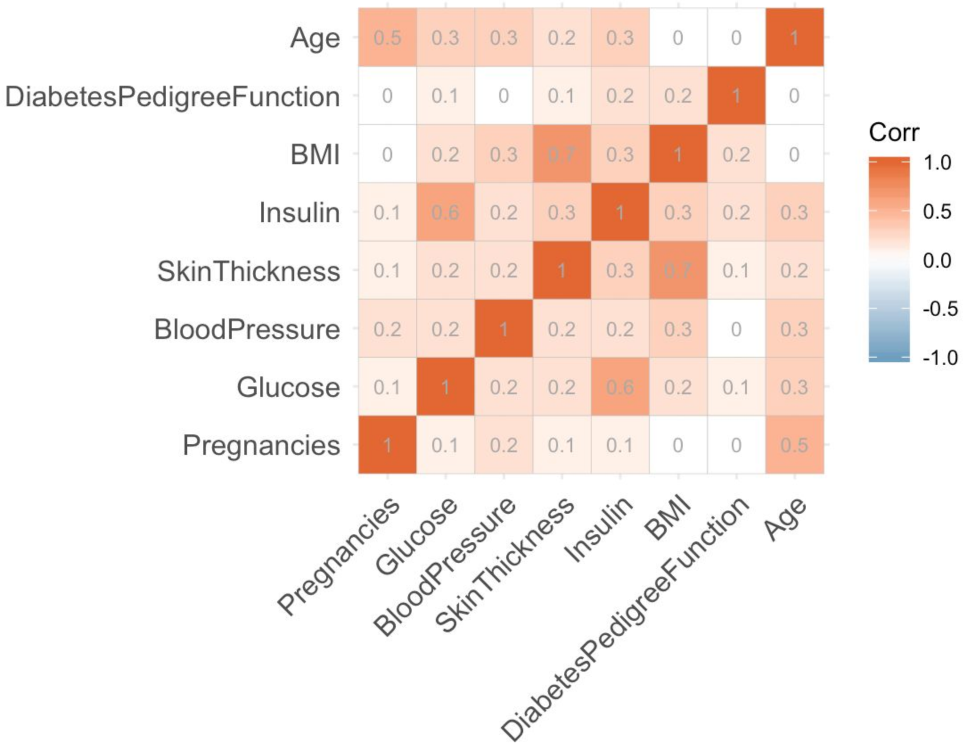
Feature Selection: Recursive Feature Elimination



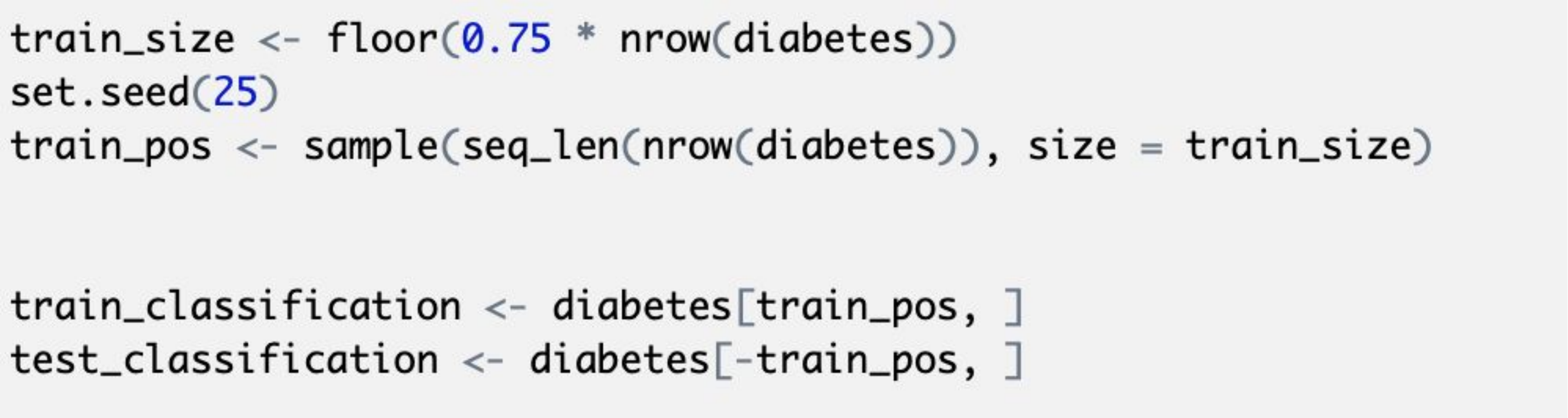
```
Recursive feature selection

Outer resampling method: Bootstrapped (2 reps)

Resampling performance over subset size:


The top 5 variables (out of 8):
   Glucose, Insulin, Age, BMI, SkinThickness

[1] "Glucose"                  "Insulin"                  "Age"
[4] "BMI"                      "SkinThickness"            "BloodPressure"
[7] "Pregnancies"              "DiabetesPedigreeFunction"
```



Correlation Plot



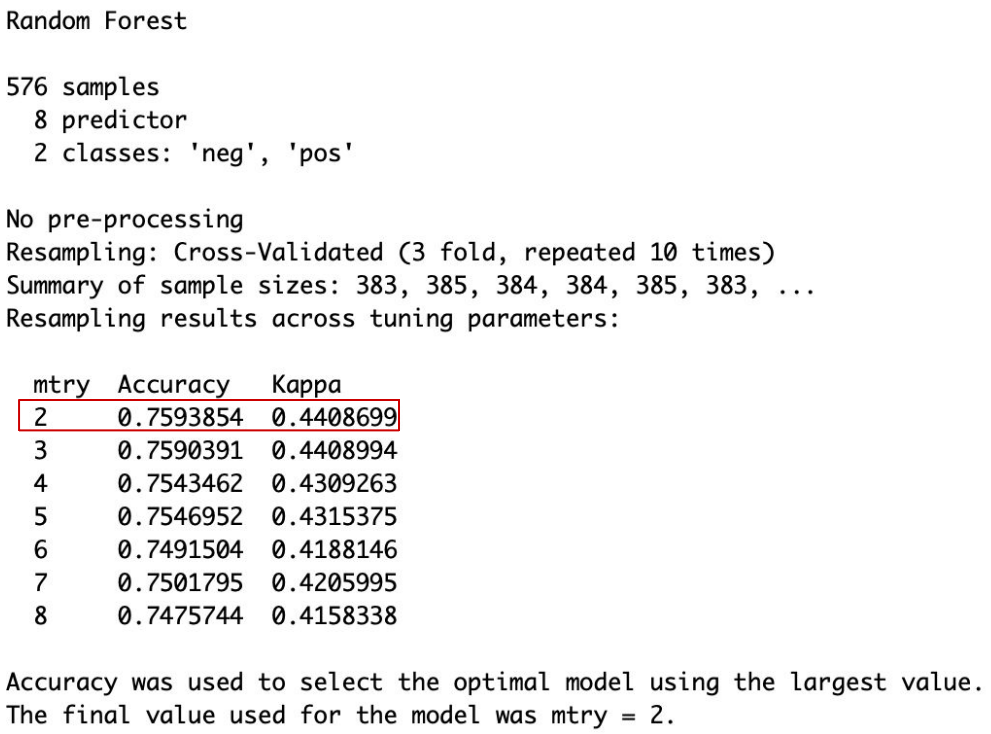
Train/Test Split



```
train_size <- floor(0.75 * nrow(diabetes))
set.seed(25)
train_pos <- sample(seq_len(nrow(diabetes)), size = train_size)


train_classification <- diabetes[train_pos, ]
test_classification <- diabetes[-train_pos, ]
```

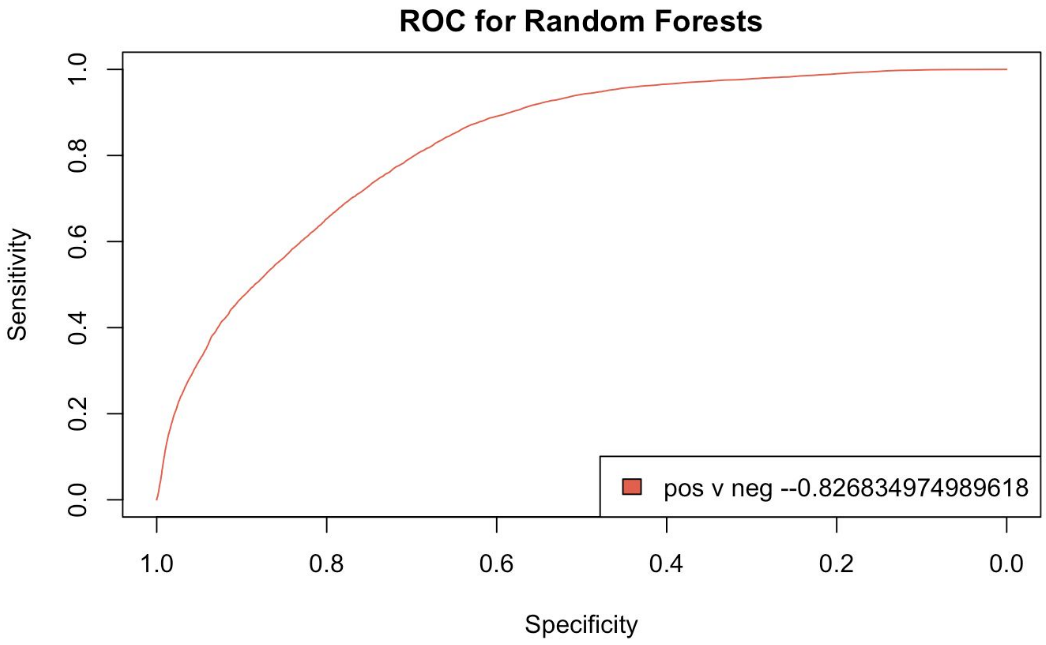
Random Forests

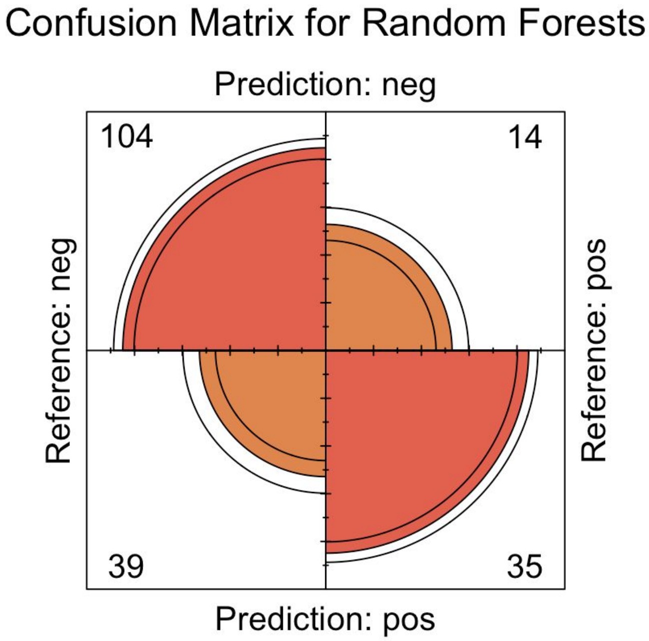
```
rf_train <- train(Outcome ~ ., data = train_classification, method = 'rf', tuneLength = 7, metric = 'Accuracy',
trControl = ctrl)
```



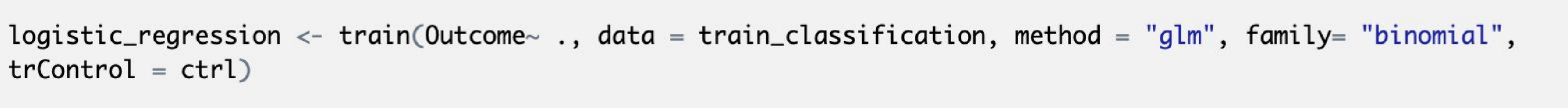
```
Random Forest

576 samples
  8 predictor
  2 classes: 'neg', 'pos'

No pre-processing
Resampling: Cross-Validated (3 fold, repeated 10 times)
Summary of sample sizes: 383, 385, 384, 384, 385, 383, ...
Resampling results across tuning parameters:

  mtry  Accuracy   Kappa
  2     0.7593854  0.4408699
  3     0.7590391  0.4408994
  4     0.7543462  0.4309263
  5     0.7546952  0.4315375
  6     0.7491504  0.4188146
  7     0.7501795  0.4205995
  8     0.7475744  0.4158338

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was mtry = 2.
```

Random Forests

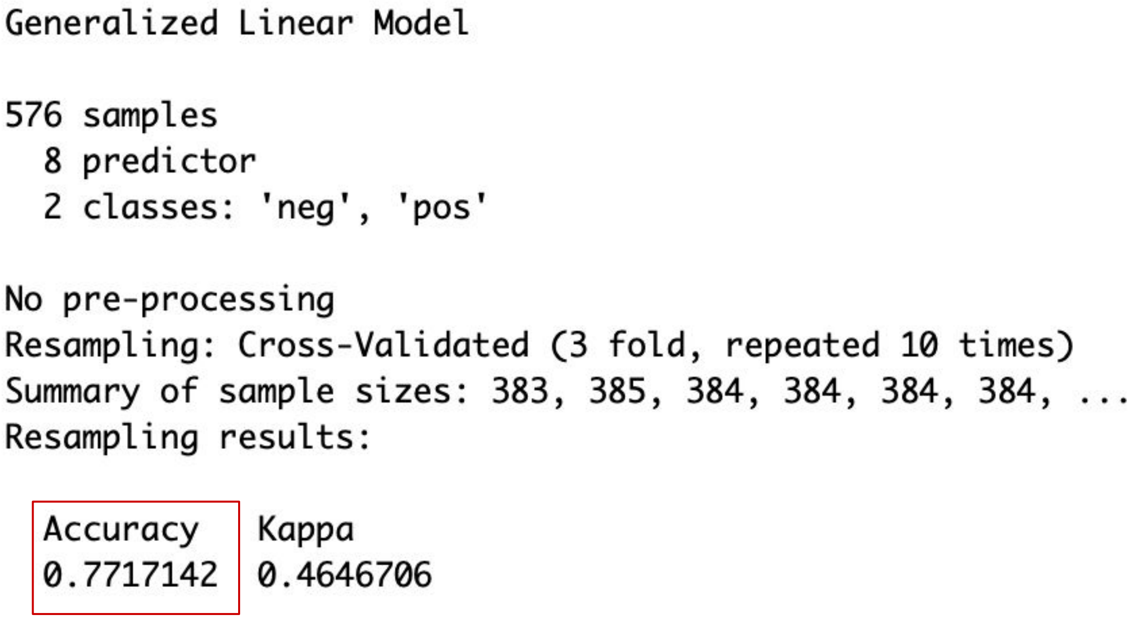




Logistic Regression

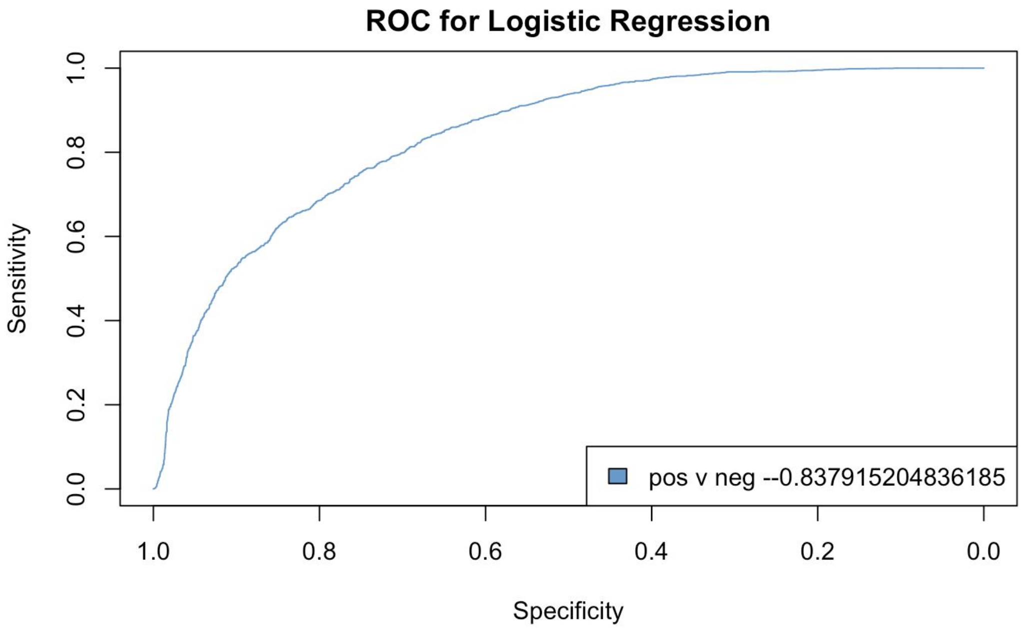


```
logistic_regression <- train(Outcome~ ., data = train_classification, method = "glm", family= "binomial",
trControl = ctrl)
```

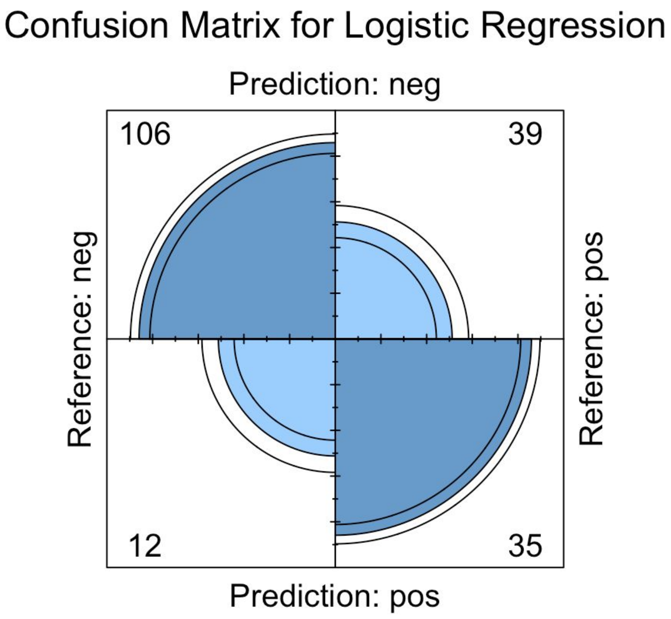


```
Generalized Linear Model

576 samples
  8 predictor
  2 classes: 'neg', 'pos'

No pre-processing
Resampling: Cross-Validated (3 fold, repeated 10 times)
Summary of sample sizes: 383, 385, 384, 384, 384, 384, ...
Resampling results:

  Accuracy   Kappa
  0.7717142  0.4646706
```

Logistic Regression

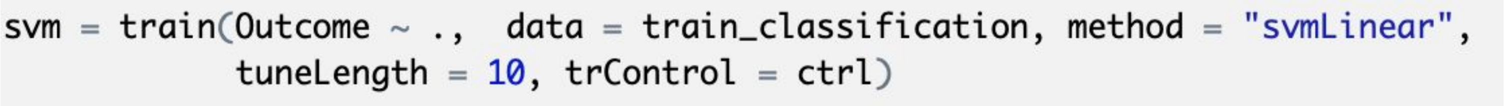


ROC for Logistic Regression

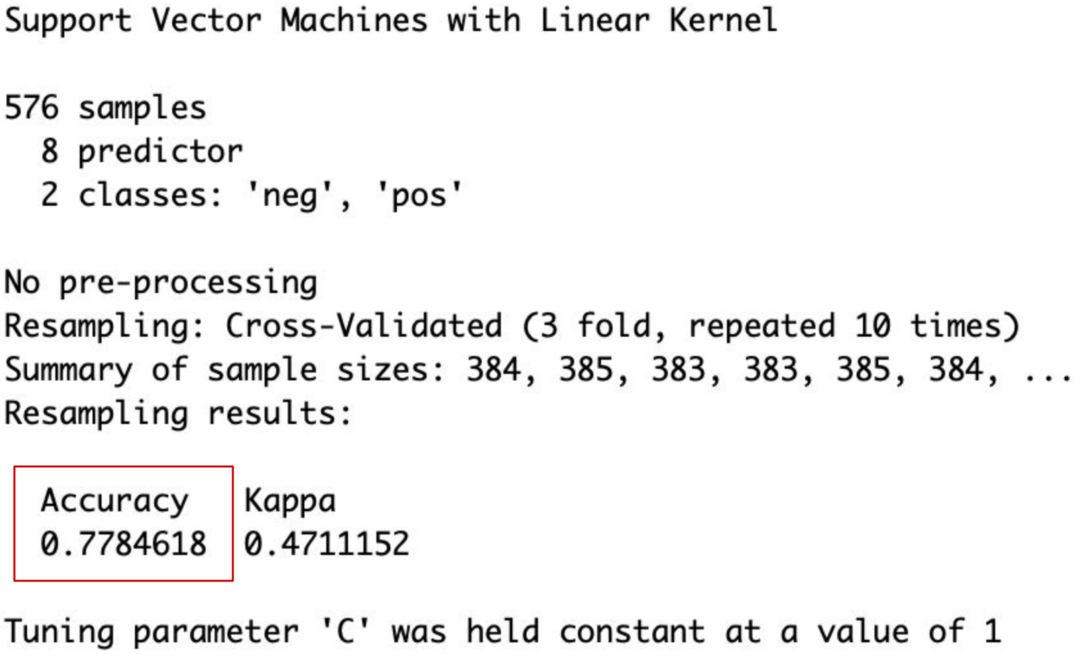


Confusion Matrix for Logistic Regression

Support Vector Machine

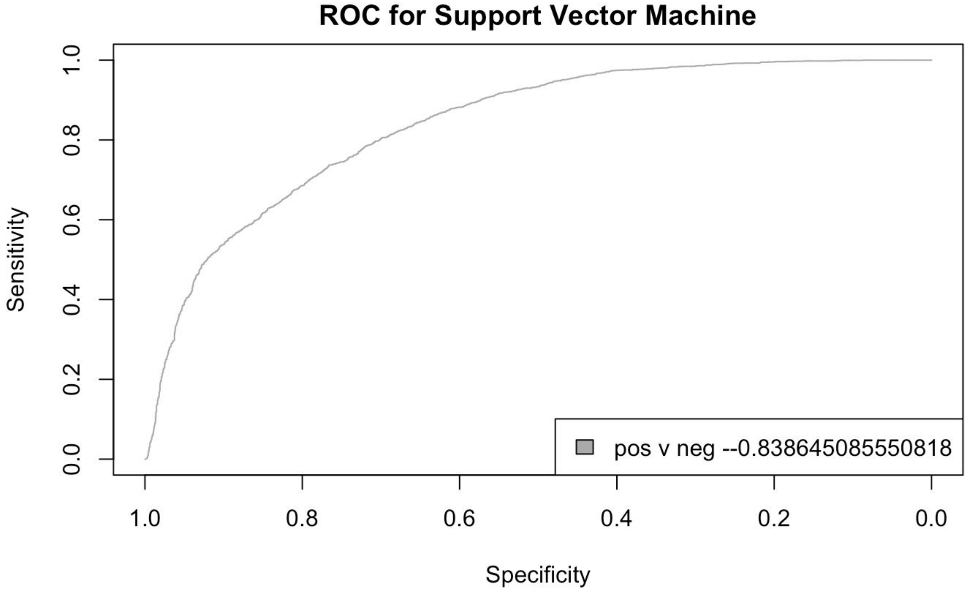


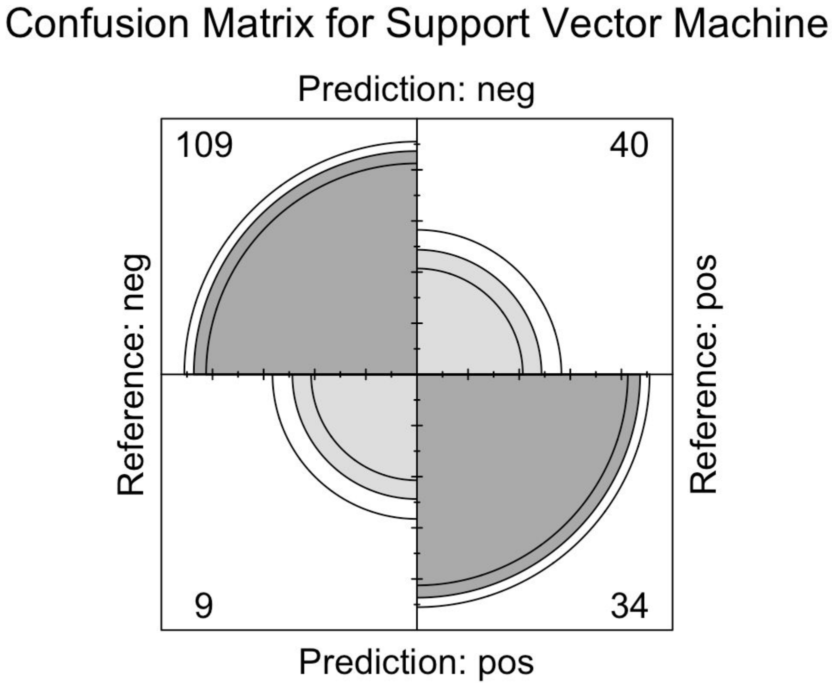
```r
svm = train(Outcome ~ .,  data = train_classification, method = "svmLinear",
            tuneLength = 10, trControl = ctrl)
```



```
Support Vector Machines with Linear Kernel

576 samples
  8 predictor
  2 classes: 'neg', 'pos'

No pre-processing
Resampling: Cross-Validated (3 fold, repeated 10 times)
Summary of sample sizes: 384, 385, 383, 383, 385, 384, ...
Resampling results:

  Accuracy   Kappa
  0.7784618  0.4711152

Tuning parameter 'C' was held constant at a value of 1
```

Support Vector Machine


ROC for Support Vector Machine


Confusion Matrix for Support Vector Machine

Conclusions

- All of the models had mediocre prediction accuracy - perhaps because of issues with the data set (uneven split of positive and negative) or because the variables are not good enough predictors of whether or not a patient has diabetes
- Out of the three SVM appeared to be the best