ADOPTABLE DOGS

By: Shannon Coakley,
Jessia Miyasato, and
Emily Miyashiro

3.1 million

Dogs enter shelters each year.

2 million

Of those dogs are adopted each year.

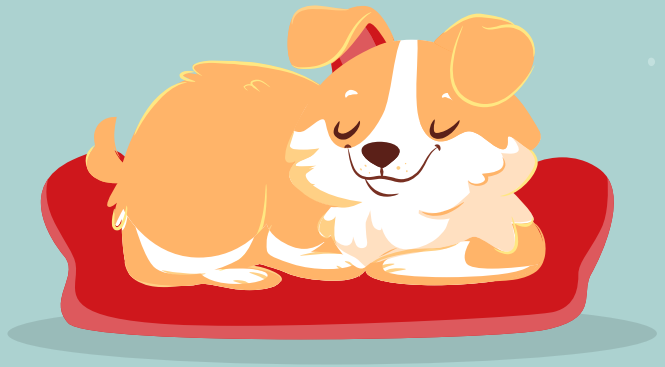
34%

Of dogs were purchased from breeders.

Prediction Goal

How likely is a dog to be adopted?

Allow shelters to predict available capacity based on how quickly a dog is to be adopted.



Raw Data

- Id: Unique identification code for the dogs
- Organization id: Unique identification code for shelter
- URL: link to dogs on Petfinder
- Type: type of animal (all dog)
- Special: species of animal (all dog)
- Breeds: primary, secondary, and tertiary breeds for each dog *
- Colors: primary, secondary, and tertiary colors of each dog *
- Age: Age of dog (Baby, Young, Adult, or Senior) *
- Gender: gender of the dog (male or female) *
- Coat: coat pattern of dog (Curly, long, medium, none, short, wire) *
- Size: how big the dog is (Small, Medium, Large, Extra Large) *
- Environment: what type of environment the dog came from
- Name: Name of the dog
- Description: Brief description of the dog
- Organization_animal_id: The ID of the dog from the shelter organization
- Photos: link to photos of the dogs
- Status Change Date: Date status became “adopted” or “adoptable”
- Neutered: Status of if the dog is spayed or neutered *
- Status: Whether the dog had been adopted or is available for adoption *
- Attributes: Whether the dog was house trained or have special needs

Summary of Cleaning

```
adogs1 <- read_csv("datasets/pet-adoption.csv")

adogs1[c('primary_breed', 'other_breeds')] <- str_split_fixed(adogs1$breeds, ',', 2)
adogs1[c('p','p1_breed' )] <- str_split_fixed(adogs1$primary_breed, ':', 2)
adogs1$p1_breed<-gsub("'","",adogs1$p1_breed)
adogs1[c('breed', 'breed_other')] <- str_split_fixed(adogs1$p1_breed, '/', 2)


adogs1[c('primary_color', 'other_colors')] <- str_split_fixed(adogs1$colors, ',', 2)
adogs1$primary_color<-gsub("'","",adogs1$primary_color)
adogs1[c('p1','p1_color' )] <- str_split_fixed(adogs1$primary_color, ':', 2)
adogs1[c('c', 'c_other')] <- str_split_fixed(adogs1$p1_color, '/', 2)
adogs1[c('color', 'c1_other')] <- str_split_fixed(as.character(adogs1$c), '\\s*\\(|\\)', 2)

adogs1[c('spayed_n', 'other')] <- str_split_fixed(adogs1$attributes, ',', 2)
adogs1[c('n','neutered' )] <- str_split_fixed(adogs1$spayed_n, ':', 2)
adogs1$neutered<-gsub("'","",adogs1$neutered)

view(adogs1)
```

```
glimpse(adogs1)

adogs1 <-
  adogs1 %>%
  mutate(age = as.factor(age),
         gender = as.factor(gender),
         color = as.factor(color),
         breed = as.factor(breed),
         coat = as.factor(coat),
         size = as.factor(size),
         neutered = as.factor(neutered),
         adopted = as.factor(status))

#There are 90 factors for breed, I am going to make this the top 12. Also recoded adopted variable to make it easier to interpret.

adogs1 <-
  adogs1 %>%
  mutate(breed = fct_lump_n(breed, n = 12),
         adopted = fct_recode(adopted,
                              "not adopted" = "adoptable",
                              "adopted" = "adopted"))

#Also needed to create a dummy variable in order to calculate RMSE

adopted_dummy = ifelse(adogs1$adopted == "adopted", 1, 0)

adogs1 <- cbind(adogs1, adopted_dummy)
```

- Create tidy data
- Factor variables
- Create necessary dummy variables

Summary Statistics

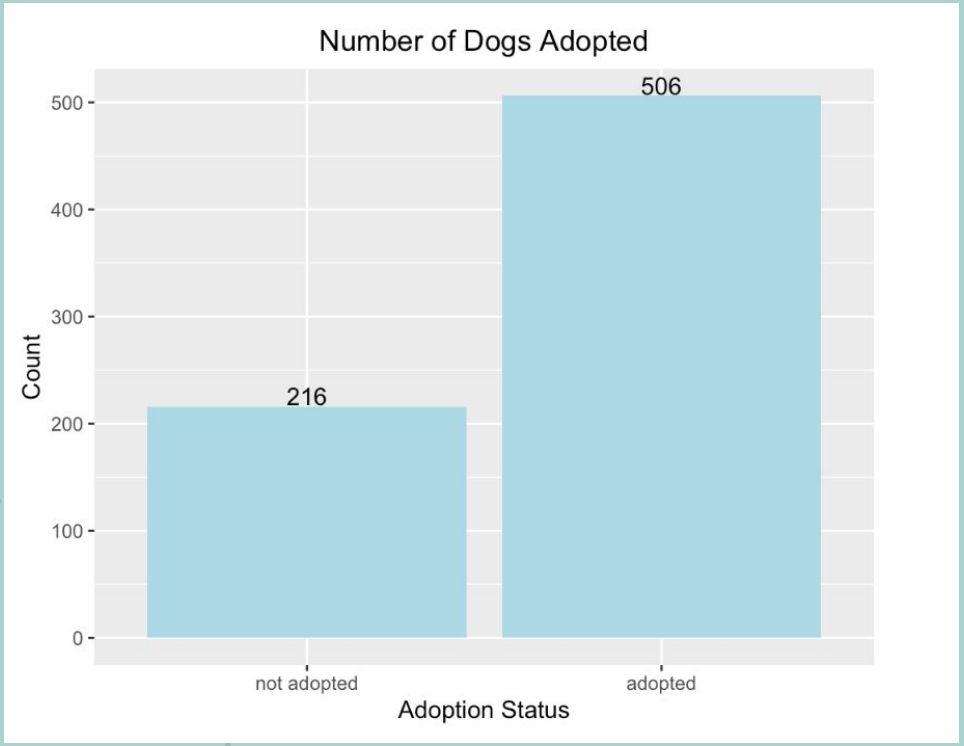
Variable	N	Mean
id	722	48552589.717
type	722	
... Dog	722	100%
species	722	
... Dog	722	100%
age	722	
... Adult	283	39.2%
... Baby	196	27.1%
... Senior	59	8.2%
... Young	184	25.5%
gender	722	
... Female	340	47.1%
... Male	382	52.9%
size	722	
... Extra Large	7	1%
... Large	163	22.6%
... Medium	405	56.1%
... Small	147	20.4%
coat	722	
... Curly	1	0.1%
... Long	17	2.4%
... Medium	87	12%
... None	325	45%
... Short	286	39.6%
... Wire	6	0.8%

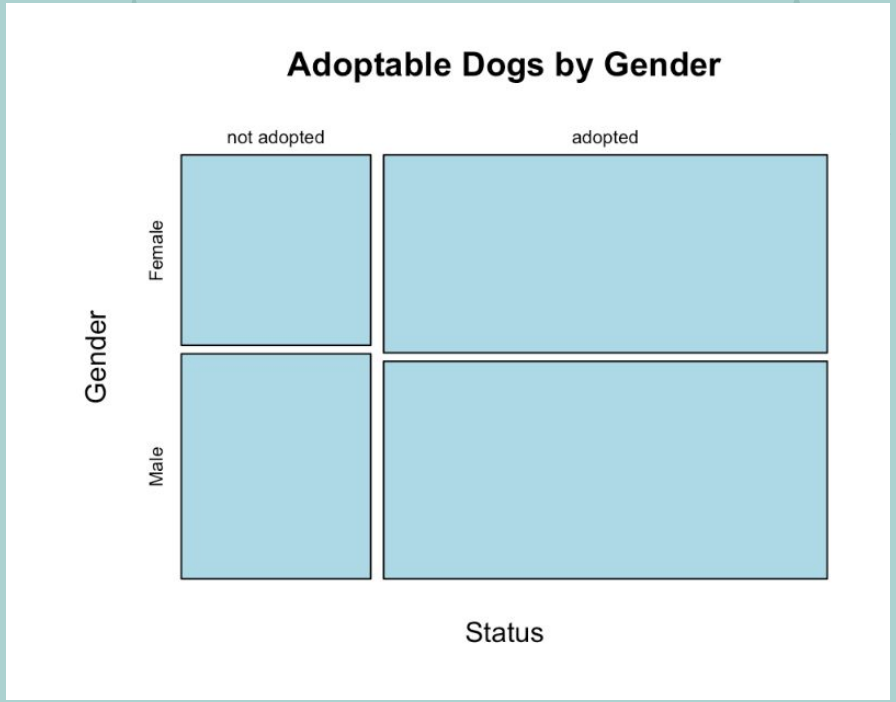
status	722	
... adoptable	216	29.9%
... adopted	506	70.1%
distance	722	
... None	722	100%
p	722	
... {'primary'	722	100%
breed	722	
... Australian Cattle Dog	17	2.4%
... Beagle	22	3%
... Chihuahua	46	6.4%
... Dachshund	19	2.6%
... German Shepherd Dog	35	4.8%
... Hound	30	4.2%
... Husky	21	2.9%
... Labrador Retriever	68	9.4%
... Mixed Breed	55	7.6%
... Pit Bull Terrier	53	7.3%
... Shepherd	29	4%
... Terrier	57	7.9%
... Other	270	37.4%

color	722	
... Apricot	15	2.1%
... Bicolor	22	3%
... Black	131	18.1%
... Brindle	41	5.7%
... Brown	29	4%
... Golden	11	1.5%
... Gray	11	1.5%
... Harlequin	1	0.1%
... Merle	10	1.4%
... None	309	42.8%
... Red	27	3.7%
... Tricolor	42	5.8%
... White	50	6.9%
... Yellow	23	3.2%
neutered	722	
... False	187	25.9%
... True	535	74.1%
adopted	722	
... not adopted	216	29.9%
... adopted	506	70.1%

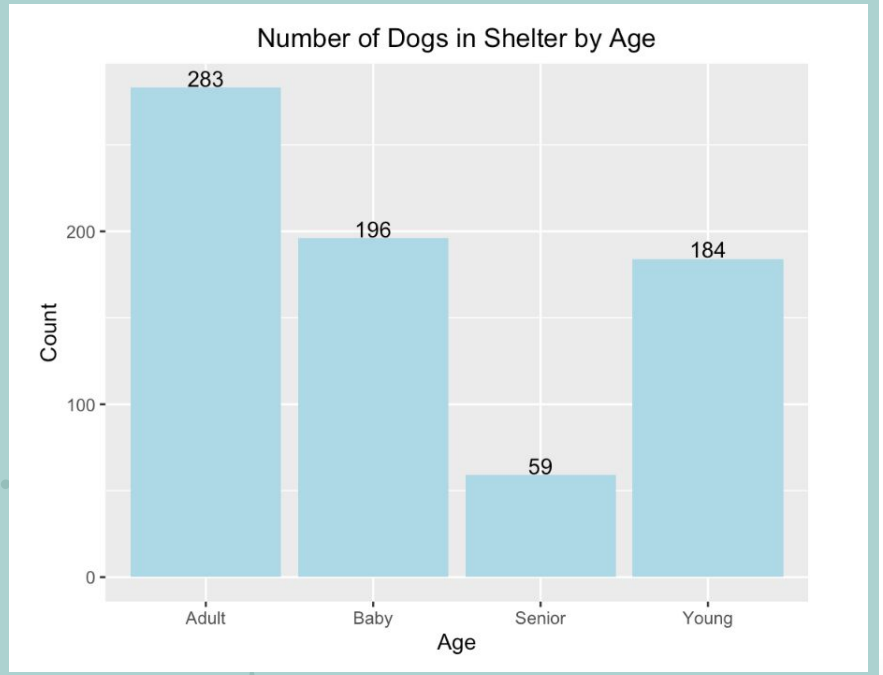
```
> dim(adogs1)
[1] 722  46
```

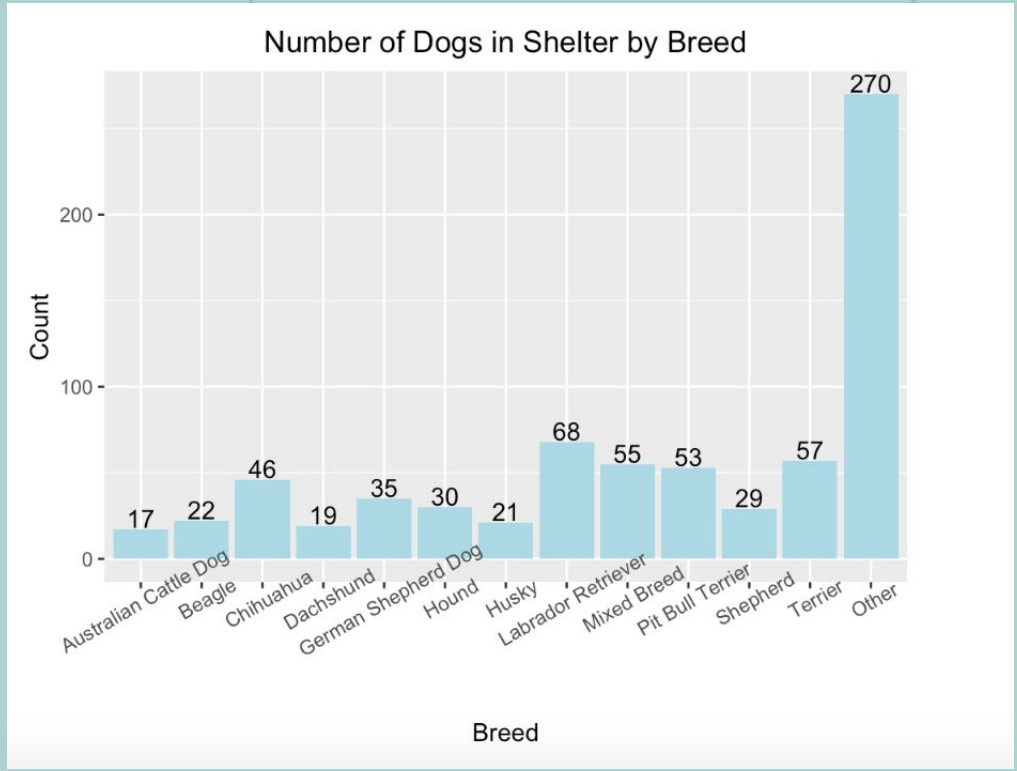
Summary Plots



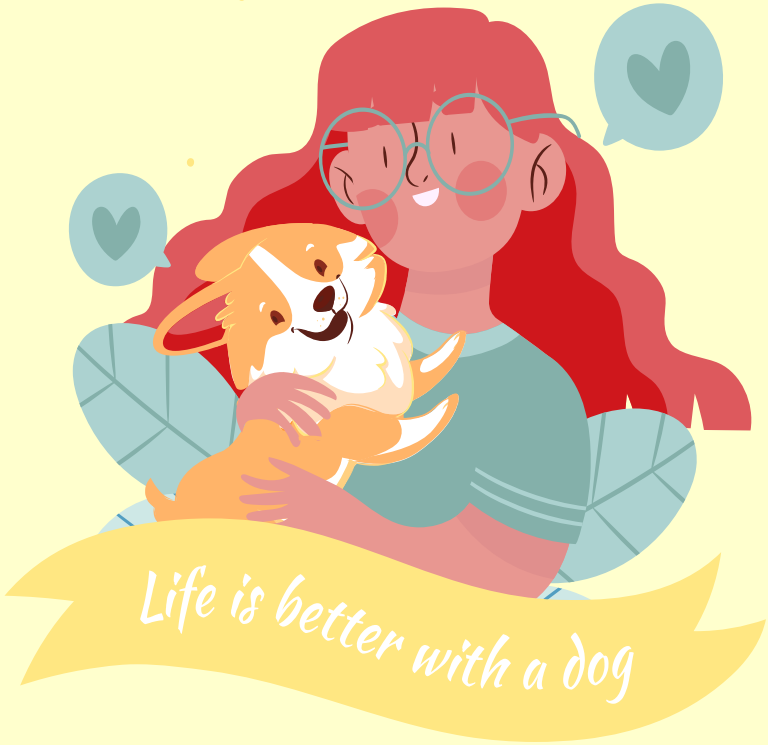


Summary Plots





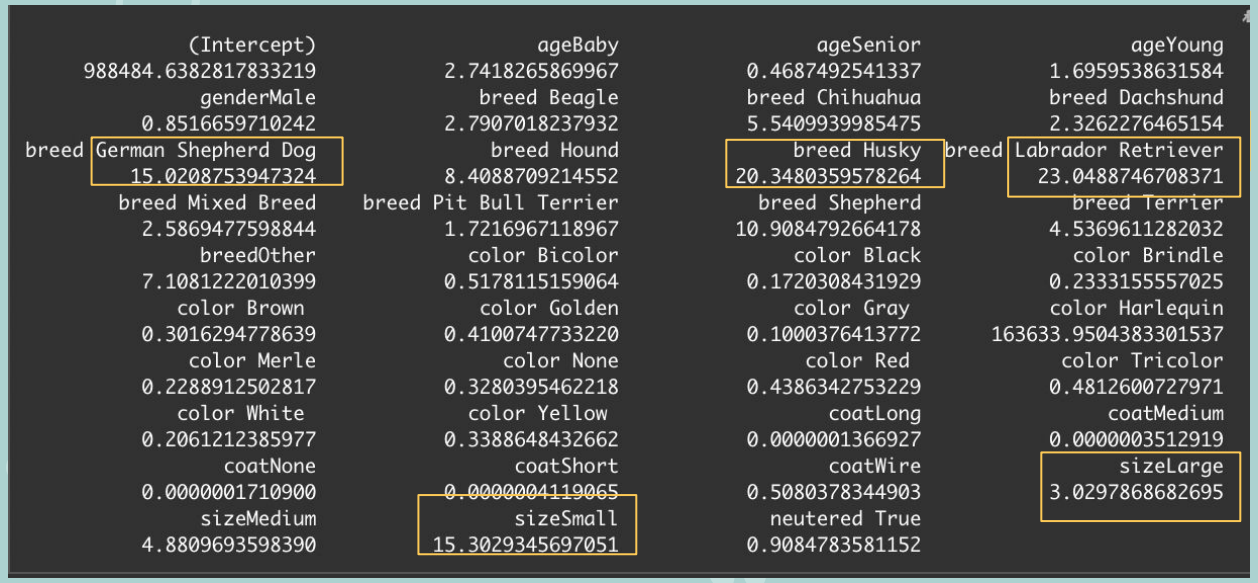
01 Logistic Regression



```r
adogs1_split <- initial_split(adogs1)
adogs1_train <- training(adogs1_split)
adogs1_test <- testing(adogs1_split)

dogs_logit <- glm(adopted ~ age + gender + breed + color + coat + size + neutered,
                  data=adogs1_train,
                  family = binomial)
```

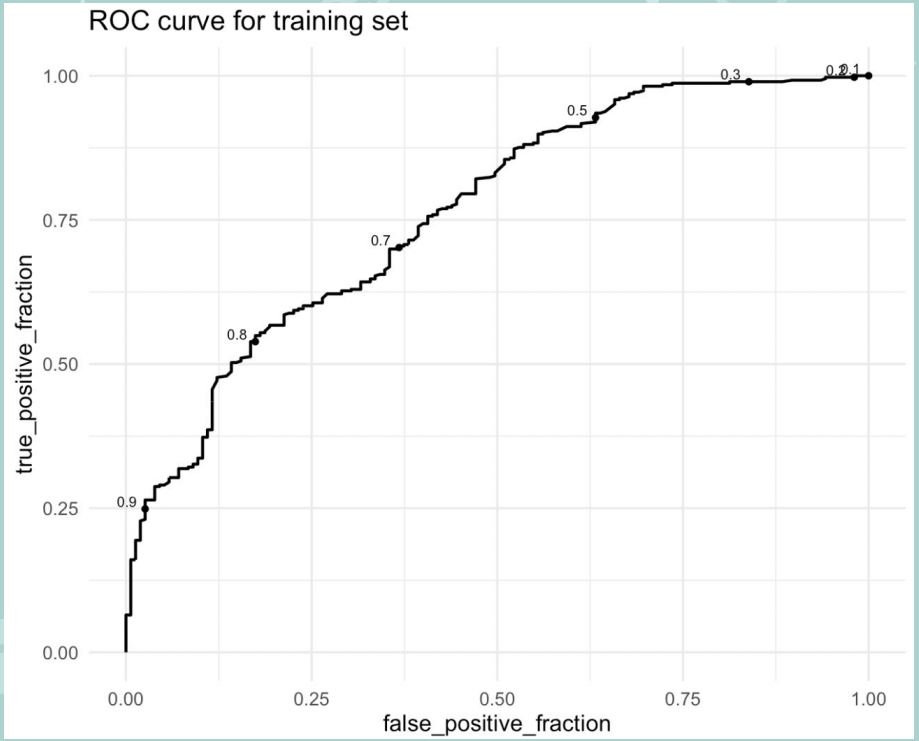
```
Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)                13.80393 1455.39862   0.009 0.992432
ageBaby                     1.00862    0.29976   3.365 0.000766 ***
ageSenior                  -0.75769    0.40042  -1.892 0.058461 .
ageYoung                    0.52825    0.28414   1.859 0.063014 .
genderMale                 -0.16056    0.22701  -0.707 0.479391
breed Beagle                1.02629    0.92899   1.105 0.269274
breed Chihuahua             1.71217    0.90460   1.893 0.058392 .
breed Dachshund             0.84425    0.97976   0.862 0.388861
breed German Shepherd Dog   2.70944    0.90743   2.986 0.002828 **
breed Hound                 2.12929    0.92585   2.300 0.021458 *
breed Husky                 3.01298    1.06244   2.836 0.004570 **
breed Labrador Retriever    3.13762    0.87019   3.606 0.000311 ***
breed Mixed Breed           0.95048    0.77633   1.224 0.220827
breed Pit Bull Terrier      0.54331    0.78142   0.695 0.486878
breed Shepherd              2.38954    0.88107   2.712 0.006686 **
breed Terrier               1.51226    0.82725   1.828 0.067541 .
breedOther                  1.96124    0.73740   2.660 0.007821 **
color Bicolor              -0.65814    1.29870  -0.507 0.612314
color Black                -1.76008    1.17951  -1.492 0.135642
color Brindle              -1.45536    1.24075  -1.173 0.240807
color Brown                -1.19856    1.28343  -0.934 0.350373
color Golden               -0.89142    1.44017  -0.619 0.535939
color Gray                 -2.30221    1.43417  -1.605 0.108438
color Harlequin            12.00539 1455.39803   0.008 0.993418
color Merle                -1.47451    1.45408  -1.014 0.310560
color None                 -1.11462    1.18056  -0.944 0.345094
color Red                  -0.82409    1.32138  -0.624 0.532851
color Tricolor             -0.73135    1.27650  -0.573 0.566691
color White                -1.57929    1.21435  -1.301 0.193420
color Yellow               -1.08215    1.37186  -0.789 0.430215
coatLong                  -15.80553 1455.39774  -0.011 0.991335
coatMedium                -14.86165 1455.39764  -0.010 0.991853
coatNone                  -15.58108 1455.39764  -0.011 0.991458
coatShort                 -14.70247 1455.39763  -0.010 0.991940
coatWire                   -0.67720 1563.53367   0.000 0.999654
sizeLarge                   1.10849    0.99077   1.119 0.263216
sizeMedium                  1.58534    0.98253   1.614 0.106628
sizeSmall                   2.72804    1.03434   2.637 0.008353 *
neutered True              -0.09598    0.26544  -0.362 0.717651
```

Exponentiated coefficients



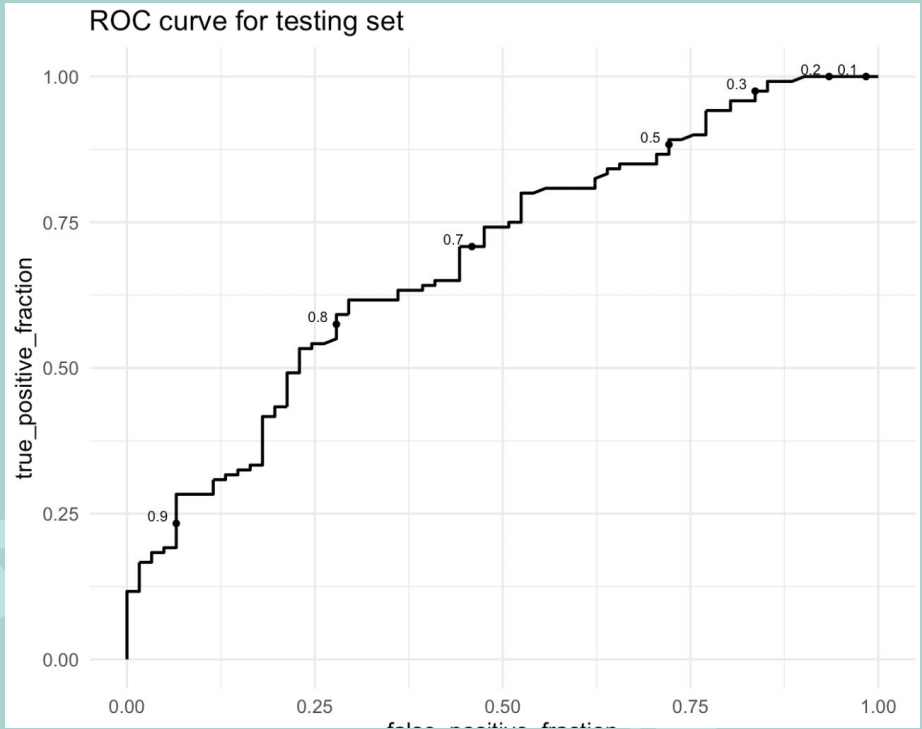
(Intercept)	ageBaby	ageSenior	ageYoung
988484.6382817833219	2.7418265869967	0.4687492541337	1.6959538631584
genderMale	breed Beagle	breed Chihuahua	breed Dachshund
0.8516659710242	2.7907018237932	5.5409939985475	2.3262276465154
breed German Shepherd Dog	breed Hound	breed Husky	breed Labrador Retriever
15.0208753947324	8.4088709214552	20.3480359578264	23.0488746708371
breed Mixed Breed	breed Pit Bull Terrier	breed Shepherd	breed Terrier
2.5869477598844	1.7216967118967	10.9084792664178	4.5369611282032
breedOther	color Bicolor	color Black	color Brindle
7.1081222010399	0.5178115159064	0.1720308431929	0.2333155557025
color Brown	color Golden	color Gray	color Harlequin
0.3016294778639	0.4100747733220	0.1000376413772	163633.9504383301537
color Merle	color None	color Red	color Tricolor
0.2288912502817	0.3280395462218	0.4386342753229	0.4812600727971
color White	color Yellow	coatLong	coatMedium
0.2061212385977	0.3388648432662	0.0000001366927	0.0000003512919
coatNone	coatShort	coatWire	sizeLarge
0.0000001710900	0.0000004119065	0.5080378344903	3.0297868682695
sizeMedium	sizeSmall	neutered True	
4.8809693598390	15.3029345697051	0.9084783581152	

- Small dogs have a 1430% probability of being adopted relative to extra large dogs.
- Large dogs have 203% probability of being adopted
- Popular dog breeds like labrador retrievers, german shepherds, and huskies have over a 1400% probability of being adopted relative to the australian cattle dog.
- Less popular breeds like pit bulls have a 72% probability of being adopted relative to australian cattle dogs




```r
scores_train <- predict(dogs_logit,
                        type = "response",
                        data = adogs1_train)

scores_test <- predict(dogs_logit, adogs1_test,
                       type = "response",
                       se.fit = FALSE,
                       dispersion = NULL,
                       terms = NULL,
                       na.action = na.pass)
```

```r
results_train <- ggplot(results_train,
            aes(m = prob_event, d = true_class)) +
  geom_roc(labelsize = 3.5,
           cutoffs.at =
            c(0.9,0.8,0.7,0.5,0.3,0.2,0.1)) +
  theme_minimal(base_size = 16)
print(results_train + ggtitle("ROC curve for training set"))
```




```
results_test <- ggplot(results_test,
            aes(m = prob_event, d = true_class)) +
  geom_roc(labelsize = 3.5,
           cutoffs.at =
             c(0.9,0.8,0.7,0.5,0.3,0.2,0.1)) +
  theme_minimal(base_size = 16)
print(results_test + ggtitle("ROC curve for testing set"))
```

```
calc_auc(results_train)
```

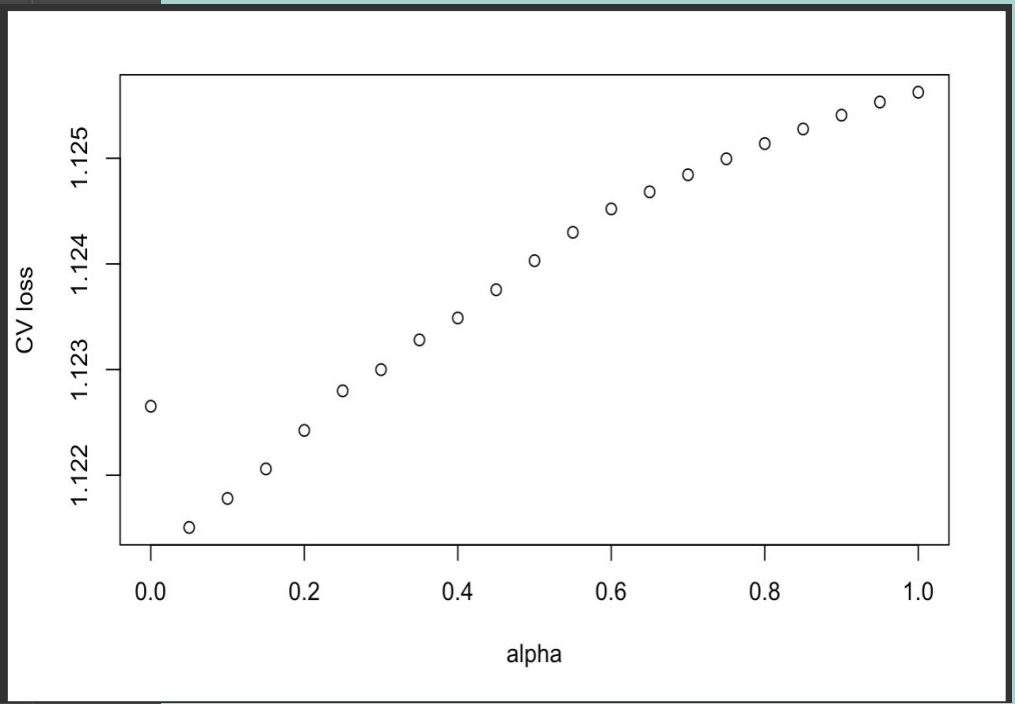
AUC <dbl>
0.7622764

```
calc_auc(results_test)
```

AUC <dbl>
0.6891393

02 Ridge Regression

```r
adogs1_split <- initial_split(adogs1)
adogs1_train <- training(adogs1_split)
adogs1_test <- testing(adogs1_split)

enet_mod1 <- cva.glmnet(adopted ~ age + gender + color + breed + coat + size + neutered,
                        data = adogs1,
                        alpha = seq(0,1, by = 0.05),
                        family = "binomial")

get_alpha <- function(fit) {
  alpha <- fit$alpha
  error <- sapply(fit$modlist,
                  function(mod) {min(mod$cvm)})
  alpha[which.min(error)]
}

# Get all parameters.
get_model_params <- function(fit) {
  alpha <- fit$alpha
  lambdaMin <- sapply(fit$modlist, `[[`, "lambda.min")
  lambdaSE <- sapply(fit$modlist, `[[`, "lambda.1se")
  error <- sapply(fit$modlist, function(mod) {min(mod$cvm)})
  best <- which.min(error)
  data.frame(alpha = alpha[best], lambdaMin = lambdaMin[best],
             lambdaSE = lambdaSE[best], eror = error[best])
}

# extract the best alpha value and model parameters
best_alpha <- get_alpha(enet_mod1)
print(best_alpha)
get_model_params(enet_mod1)

minlossplot(enet_mod1,
            cv.type = "min")
```



alpha <dbl>	lambdaMin <dbl>	lambdaSE <dbl>	eror <dbl>
0.05	0.06873802	0.3342456	1.121505

1 row

```r
dog_mod <- cv.glmnet(adopted ~ age + gender + color + breed + coat + size + neutered,
                     data = adogs1_train,
                     alpha = 0,
                     family = "binomial")

coefpath(dog_mod)
print(dog_mod$lambda.min)
#
print(dog_mod$lambda.1se)

# print coefficient using lambda.min
coef(dog_mod, s = dog_mod$lambda.min) %>%
  round(3)

# print coefficient using lambda.1se
coef(dog_mod, s = dog_mod$lambda.1se) %>%
  round(3)

# put into coefficient vector
dog_coefs <- tibble(
  `varnames` = rownames(coef(dog_mod, s = dog_mod$lambda.1se)),
  `ridge_min` = coef(dog_mod, s = dog_mod$lambda.min) %>%
    round(3) %>% as.matrix() %>% as.data.frame(),
  `ridge_1se` = coef(dog_mod, s = dog_mod$lambda.1se) %>%
    round(3) %>% as.matrix() %>% as.data.frame()
)

print(dog_coefs)

plot(dog_mod)
```

lambda.min

```
[1] 0.122627
```

lambda.1se

```
[1] 0.8651092
```

(Intercept)	1.108
ageAdult	-0.323
ageBaby	0.416
ageSenior	-0.483
ageYoung	0.136
genderFemale	0.045
genderMale	-0.045
color Apricot	0.701
color Bicolor	0.380
color Black	-0.183
color Brindle	-0.077
color Brown	0.008
color Golden	0.091
color Gray	-0.375
color Harlequin	0.831
color Merle	-0.191
color None	-0.071
color Red	0.291
color Tricolor	0.275
color White	-0.093
color Yellow	0.250
breed American Bulldog	0.124
breed Australian Cattle Dog	-0.758
breed Australian Shepherd	1.207
breed Beagle	-0.432
breed Border Collie	0.933
breed Boxer	-0.470
breed Cattle Dog	-1.522
breed Chihuahua	0.183
breed Collie	0.728
breed Dachshund	-0.336
breed German Shepherd Dog	0.333
breed Hound	0.102
breed Husky	0.663
breed Labrador Retriever	0.741
breed Mixed Breed	-0.648
breed Pit Bull Terrier	-0.877
breed Retriever	1.165
breed Shepherd	0.221
breed Shih Tzu	0.380
breed Terrier	0.015
breedOther	-0.097

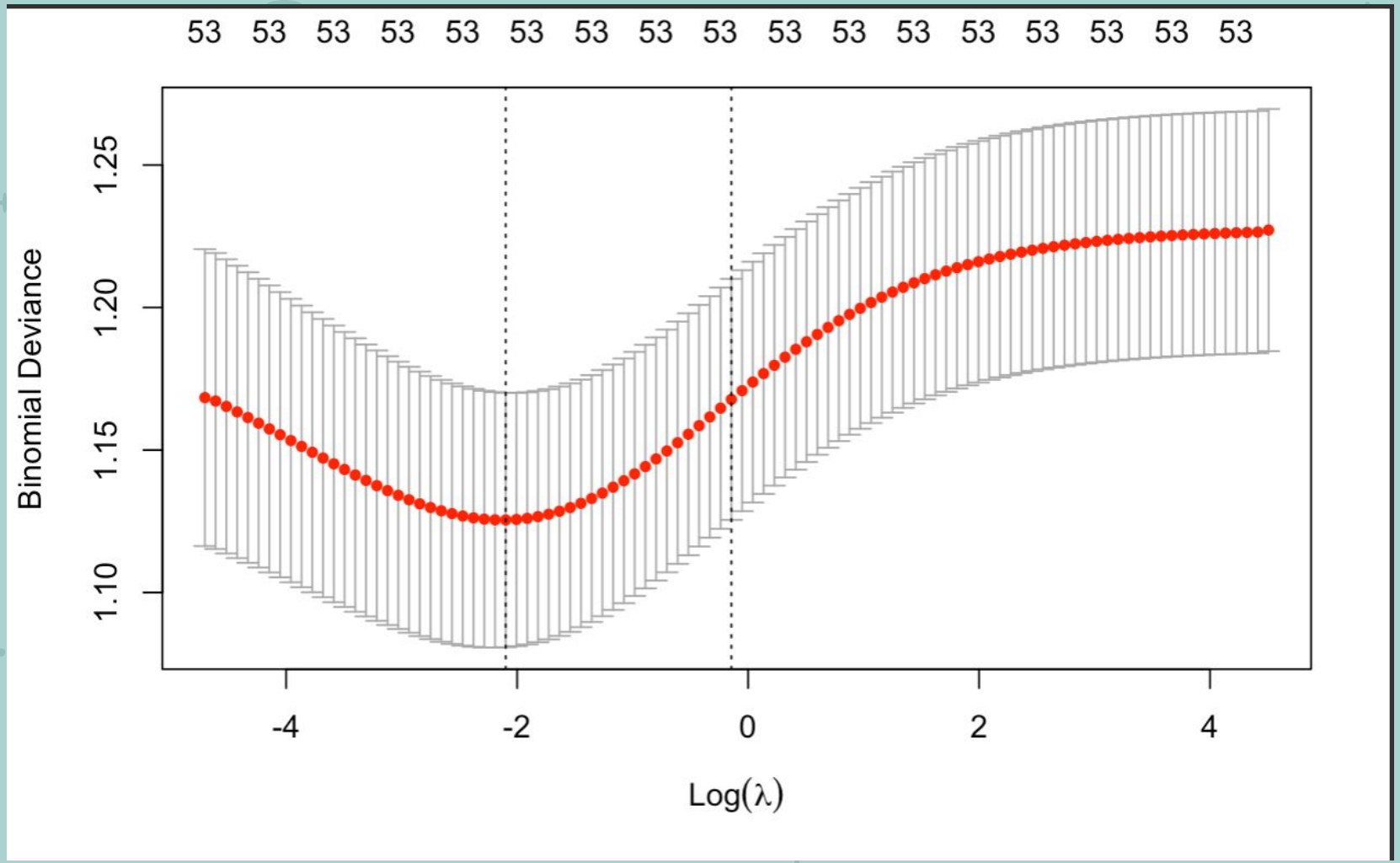
lambda.min

coatCurly	.
coatLong	-0.530
coatMedium	0.153
coatNone	-0.274
coatShort	0.216
coatWire	1.372
sizeExtra Large	-1.385
sizeLarge	-0.388
sizeMedium	-0.067
sizeSmall	0.578
neutered False	-0.038
neutered True	0.038

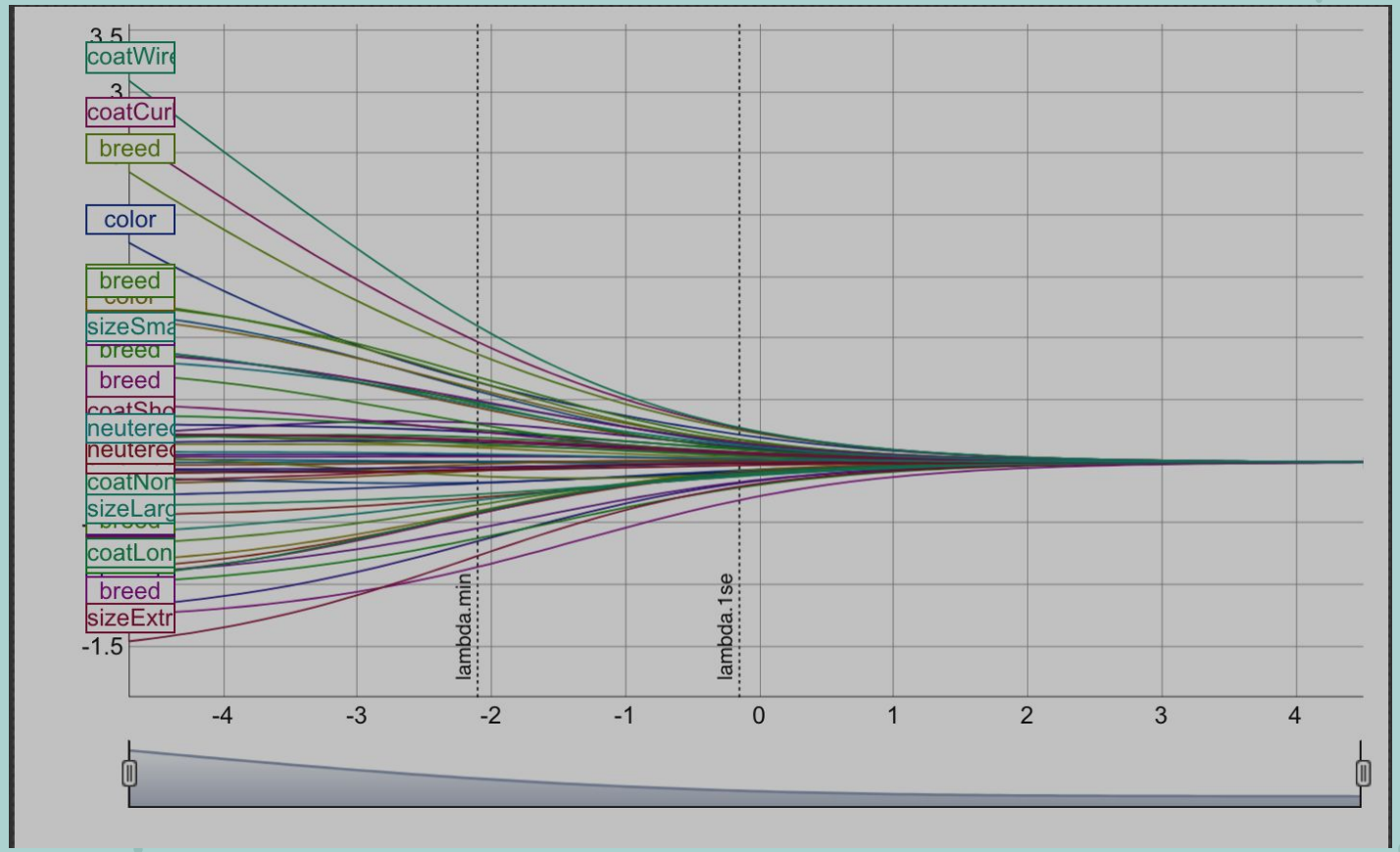
(Intercept)	0.977
ageAdult	-0.172
ageBaby	0.197
ageSenior	-0.142
ageYoung	0.057
genderFemale	0.025
genderMale	-0.025
color Apricot	0.253
color Bicolor	0.131
color Black	-0.049
color Brindle	-0.025
color Brown	0.020
color Golden	0.034
color Gray	-0.155
color Harlequin	0.337
color Merle	-0.018
color None	-0.056
color Red	0.113
color Tricolor	0.138
color White	-0.028
color Yellow	0.128
breed American Bulldog	-0.002
breed Australian Cattle Dog	-0.279
breed Australian Shepherd	0.423
breed Beagle	-0.140
breed Border Collie	0.330
breed Boxer	-0.198
breed Cattle Dog	-0.606
breed Chihuahua	0.150
breed Collie	0.273
breed Dachshund	-0.081
breed German Shepherd Dog	0.086
breed Hound	0.084
breed Husky	0.247
breed Labrador Retriever	0.296
breed Mixed Breed	-0.279
breed Pit Bull Terrier	-0.424
breed Retriever	0.416
breed Shepherd	0.088
breed Shih Tzu	0.181
breed Terrier	0.069
breedOther	-0.042

lambda.1se

coatCurly	.
coatLong	-0.171
coatMedium	0.070
coatNone	-0.142
coatShort	0.117
coatWire	0.424
sizeExtra Large	-0.613
sizeLarge	-0.167
sizeMedium	-0.015
sizeSmall	0.229
neutered False	-0.022
neutered True	0.022


53 53 53 53 53 53 53 53 53 53 53 53 53 53 53 53 53
Binomial Deviance
1.10
1.15
1.20
1.25
-4
-2
0
2
4
Log(λ)

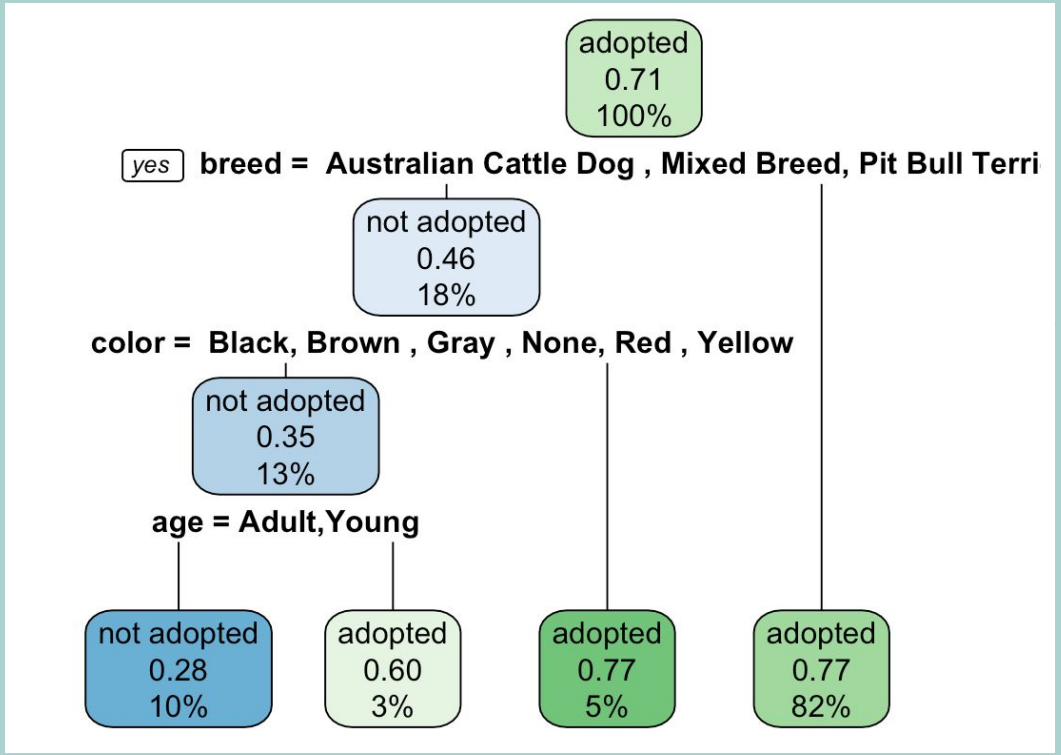
```
devtools::install_github("jaredlander/coefplot")
library('coefplot')
coefpath(dog_mod)
```



03 Decision Tree

Basic Decision Tree

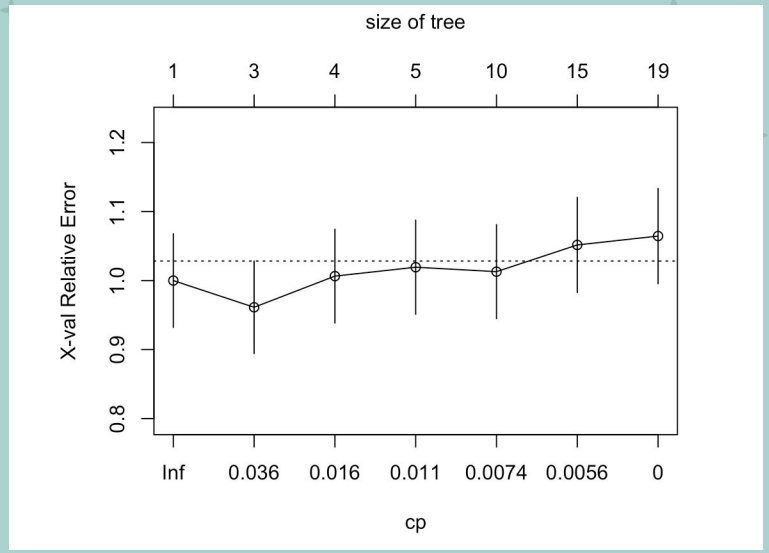
- Each node shows the predicted class, predicted probability of adoption, and the percentage of observations in that node
- Any breed that is not an Australian Cattle Dog, Mixed Breed, or Pit Bull have a 77% of being adopted




```
#code for basic tree
library('rpart')
adogs1_rpart <- rpart(adopted ~ age + gender + color + breed + coat + size + neutered,
                      data = adogs1_train)

rpart.plot(adogs1_rpart)
```

Cross-Validation

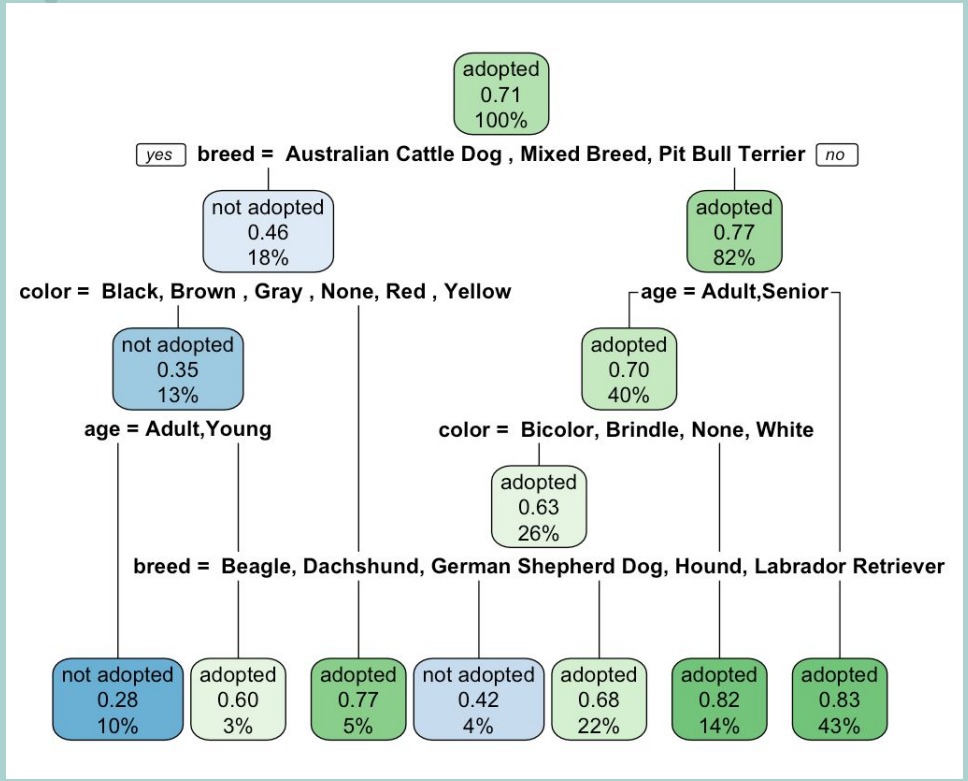
```
library('rpart')
adogs1_rpart <- rpart(adopted ~ age + gender + color + breed + coat + size + neutered,
                      data = adogs1_train,
                      control = list(cp = 0,
                                     minsplit = 10,
                                     maxdepth = 10))

adogs1_rpart$cptable
##             CP nsplit rel error    xerror       xstd
## 1 0.067741935      0 1.0000000 1.0000000 0.06784677
## 2 0.019354839      2 0.8645161 0.9612903 0.06703565
## 3 0.012903226      3 0.8451613 1.0064516 0.06797705
## 4 0.008602151      4 0.8322581 1.0193548 0.06823349
## 5 0.006451613      9 0.7677419 1.0129032 0.06810596
## 6 0.004838710     14 0.7354839 1.0516129 0.06885086
## 7 0.000000000     18 0.7161290 1.0645161 0.06908847

plotcp(adogs1_rpart)
```



- Cross validation shows the optimal number of splits to minimize error is 18.

More Complex Tree

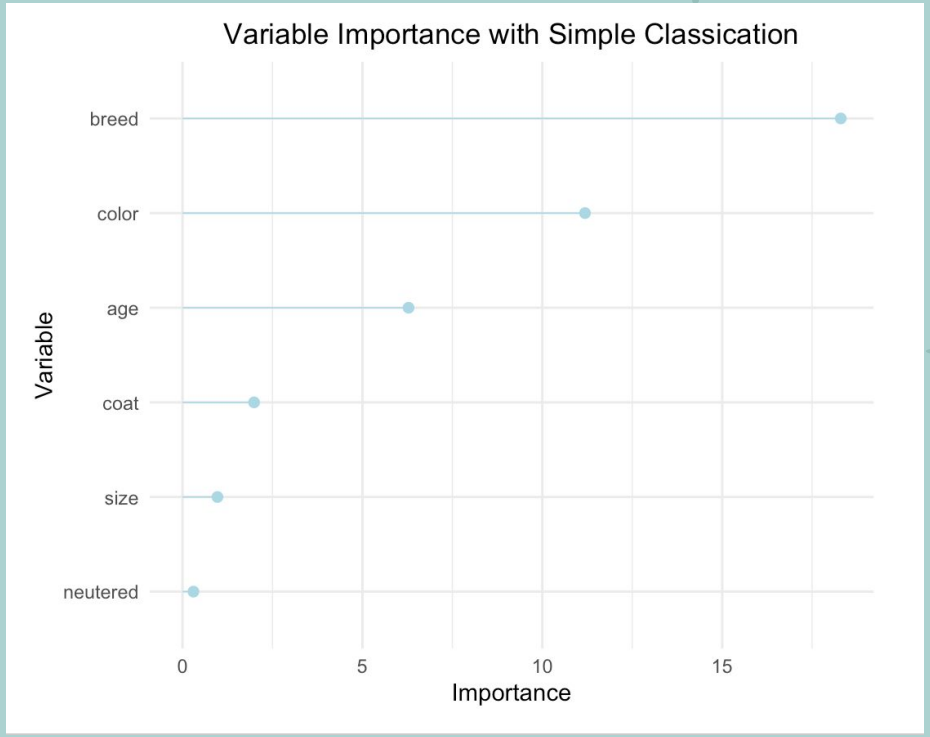
- Included the optimal split
- Same top node split, but more in depth




```
adogs1_rpart <- rpart(adopted ~ age + gender + color + breed + coat + size + neutered,
                      data = adogs1_train,
                      control = list(cp = 0,
                                     minsplit = 18,
                                     maxdepth = 4))

rpart.plot(adogs1_rpart)
```

Variable Importance

- Breed is the most important variable followed by color and age




```
adogs1_rpart$variable.importance
##      breed      color        age
## 18.2958261 11.1866218  6.2818963
##       coat       size   neutered
##  1.9883553  0.9698528  0.3025540
```

```
adogs1_rpart$variable.importance %>%
  data.frame() %>%
  rownames_to_column(var = "Feature") %>%
  rename(Overall = '.') %>%
  ggplot(aes(x = fct_reorder(Feature, Overall), y = Overall)) +
  geom_pointrange(aes(ymin = 0, ymax = Overall), color = "lightblue", size = .3) +
  theme_minimal() +
  coord_flip() +
  labs(x = "Variable", y = "Importance", title = "Variable Importance with Simple Classicatio
n") +
   theme(plot.title = element_text(hjust = 0.5))
```

Root Mean Squared Error

- RMSE = 0.5759
- This means that the model is relatively accurate at predicting adoption
- Could still be improved

```
library('Metrics')

adogs1_rpart_pred <- predict(object = adogs1_rpart, newdata = adogs1_test)

adogs1_rpart_rmse <- rmse(actual= adogs1_test$adopted_dummy,
                          predicted = adogs1_rpart_pred)

print(adogs1_rpart_rmse)
## [1] 0.5759052
```

Comparison

01

Logistic Regression

Test AUC = .6891

Model is slightly overfit since the training AUC is higher than the test AUC

02

Ridge

Finding the best lambda

03

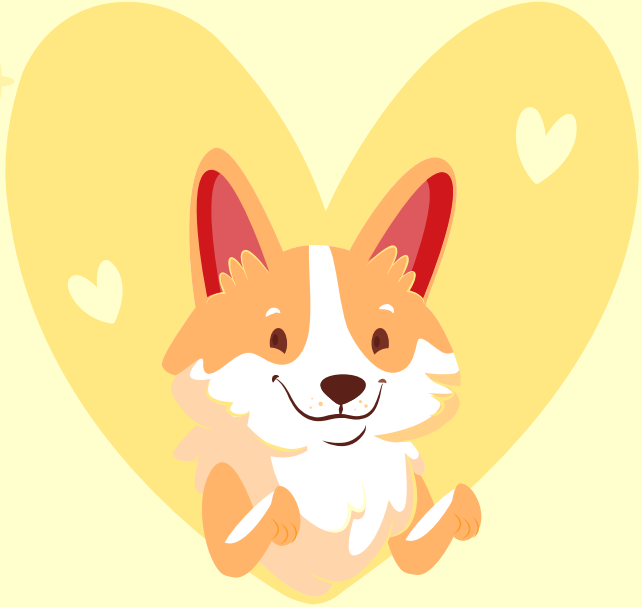
Decision Tree

RMSE = 0.5759

Model is decent at predicting which dogs with be adopted. More interpretable

Conclusion

We recommend shelters implement the use of a decision tree to predict probability of adoption. The model was pretty accurate and it is the most interpretable.



THANK YOU!