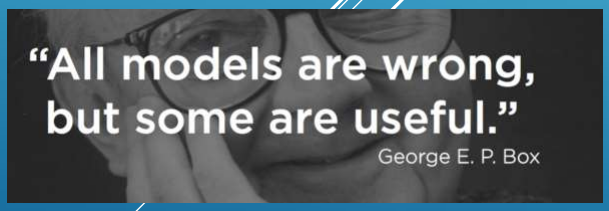
CLASSIFIERS
SUPERVISED LEARNING



LOGISTIC REGRESSION (A CLASSIFIER)

NOT REGRESSION

IS A LINEAR CLASSIFIER
AND
DISCRIMINATIVE MODEL

A classifier is linear if its decision boundary on the feature space is a linear function: positive and negative examples are separated by an **hyperplane**.

This is what a SVM does by definition without the use of the kernel trick.

Also logistic regression uses linear decision boundaries. Imagine you trained a logistic regression and obtained the coefficients β_i. You might want to classify a test record $\mathbf{x} = (x_1, \ldots, x_k)$ if $P(\mathbf{x}) > 0.5$. Where the probability is obtained with your logistic regression by:

$$P(\mathbf{x}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k)}}$$

If you work out the math you see that $P(\mathbf{x}) > 0.5$ defines a hyperplane on the feature space which separates positive from negative examples.

https://stats.stackexchange.com/questions/178522/why-knn-is-a-non-linear-classifier

https://stats.stackexchange.com/questions/12421/generative-vs-discriminative

the distribution. Two such transformations that are easy to compute are the *arcsine* and *logit* transformations. The arcsine transformation is given by

$$W_i = \sin^{-1}\sqrt{Y_i}$$

The inverse transformation for the arcsine transformation is

$$Y_i = (\sin W_i)^2$$

If r and n are small, a somewhat better arcsine transformation is given by

$$W_i = \sin^{-1}\sqrt{\frac{r_i + 3/8}{n_i + 3/4}}$$

The logit transformation is given by

$$W_i = \log\frac{Y_i}{1 - Y_i}$$

The inverse of the logit transformation is

$$Y_i = \frac{e^{W_i}}{e^{W_i} + 1} = \frac{1}{1 + e^{-W_i}}$$

For proportions the odds are given by $Y/(1 - Y)$, so the logit transformation is also known as log odds. The two transformations differ in the extent to which they stretch the tails. The logit does a little bit more stretching than the arcsine transformation. If all the values of Y_i are between about .2 and .8, then these transformations will have little effect because they operate primarily on the tails. In that case, an analysis of untransformed and transformed proportions will produce essentially the same results.

The Linear model allows us to fit one or more continuous predictor variable to a dependent continuous variable.

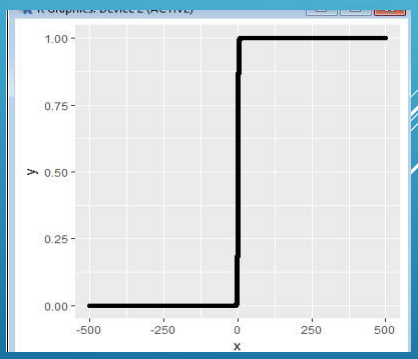
But there are scenarios where the dependent variable is not continuous. Reading an XRAY, diagnostics (benign or malignant), Admit or no-admit, fraud transaction or legitimate transaction.

We need a f(x) such that for all values of f(x) will be 0 or 1.

Very Sensitive

```
> logit(0.000001)
[1] 0.5000003
> logit(-0.000001)
[1] 0.4999997
```

```
> logit(-0.000000)
[1] 0.5
> logit(0.000000)
[1] 0.5
```




```
logit<-function(x) { 1/(1+exp(-1*x)) }
x<-seq(-500,500,by=0.01);
ggplot(data.frame(x=x,y=logit(x)),aes(x=x,y=y))+geom_point()+geom_line()
```

LOG ODDS

```
> glm.gaspx<-glm(petrol~.,data=gaspx)
> summary(glm.gaspx)

Call:
glm(formula = petrol ~ ., data = gaspx)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-122.03   -45.57   -10.66    31.53   234.95

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.773e+02  1.855e+02   2.033 0.048207 *
tax         -3.479e+01  1.297e+01  -2.682 0.010332 *
income      -6.659e-02  1.722e-02  -3.867 0.000368 ***
miles       -2.426e-03  3.389e-03  -0.716 0.477999
driver       1.336e+03  1.923e+02   6.950 1.52e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 4396.511)

    Null deviance: 588366  on 47  degrees of freedom
Residual deviance: 189050  on 43  degrees of freedom
AIC: 545.59

Number of Fisher Scoring iterations: 2
```

```
> glm.scaled.gaspx<-glm(petrol~.,data=as.data.frame(scaled.gaspx[1:48,]))
> summary(glm.scaled.gaspx)

Call:
glm(formula = petrol ~ ., data = as.data.frame(scaled.gaspx[1:48,
    ]))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.09066  -0.40732  -0.09531   0.28180   2.09988

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.879e-17  8.554e-02   0.000 1.000000
tax         -2.956e-01  1.102e-01  -2.682 0.010332 *
income      -3.414e-01  8.829e-02  -3.867 0.000368 ***
miles       -7.570e-02  1.058e-01  -0.716 0.477999
driver       6.626e-01  9.534e-02   6.950 1.52e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.3512029)

    Null deviance: 47.000  on 47  degrees of freedom
Residual deviance: 15.102  on 43  degrees of freedom
AIC: 92.711

Number of Fisher Scoring iterations: 2
```

GLM AIC AND DEVIANCE?

```
path<-"c:/Users/rkannan/rk/03062015/kirpal-story-of-data/l05"
fname<-"icu.csv"
icu<-read.csv(paste(path,fname,sep="/"),head=TRUE)
```

EXAMPLE ICU

- Generate y = 3 * x using randomly generated x
- Add some noise to y
- Use linear modeling, prove that the coeff is 3

LINEAR MODEL REVISITED