UCLA Anderson
SCHOOL of MANAGEMENT

MGMTMFE 431:

Data Analytics and Machine Learning

Topic 3: Logistic regressions, credit data, and sample selection

Spring 2019

Professor Lars A. Lochstoer

Advanced Multiple Regression Topics

a. The Logistic Regression Model
b. Interpretation of the Coefficients
c. A Simple Example
d. The Likelihood Function
e. A Simple Example (continued)
f. A More Complicated Example
g. Lift Tables
h. ROC Curves
i. Lending Club
j. Propensity Scores

c. The Logistic regression model

Suppose we have a binary dependent variable.

Examples:

1. Purchase of a product (Y=1 if purchase, Y=0 if not)
2. Click on display ad (Y=1 if click, Y=0 if not)
3. Default on loan

All of these can be formulated as a conditional prediction problem: Given X variables, what is my prediction of Y?

Since Y is binary, my predictions are probabilities that Y = 1.

UCLA Anderson SCHOOL of MANAGEMENT

c. The Logistic regression model

What is a regression model, in general?

A model for the conditional distribution of Y | X.

What is the regression line? It is E[Y|X].

If Y is binary (0,1 are the only possible values),

$$\begin{aligned} E\left[Y \mid X\right] &= \Pr\left(Y = 1 \mid X\right) \times 1 + \left(1 - \Pr\left(Y = 1 \mid X\right)\right) \times 0 \\ &= \Pr\left(Y = 1 \mid X\right) \end{aligned}$$

c. The Logistic regression model

How can we link the X variables to the probability that Y = 1?

$$\boxed{\Pr\left(Y=1\right)=\frac{\exp\left(\beta_0+\beta_1X_1+\ldots+\beta_kX_k\right)}{1+\exp\left(\beta_0+\beta_1X_1+\ldots+\beta_kX_k\right)}}$$

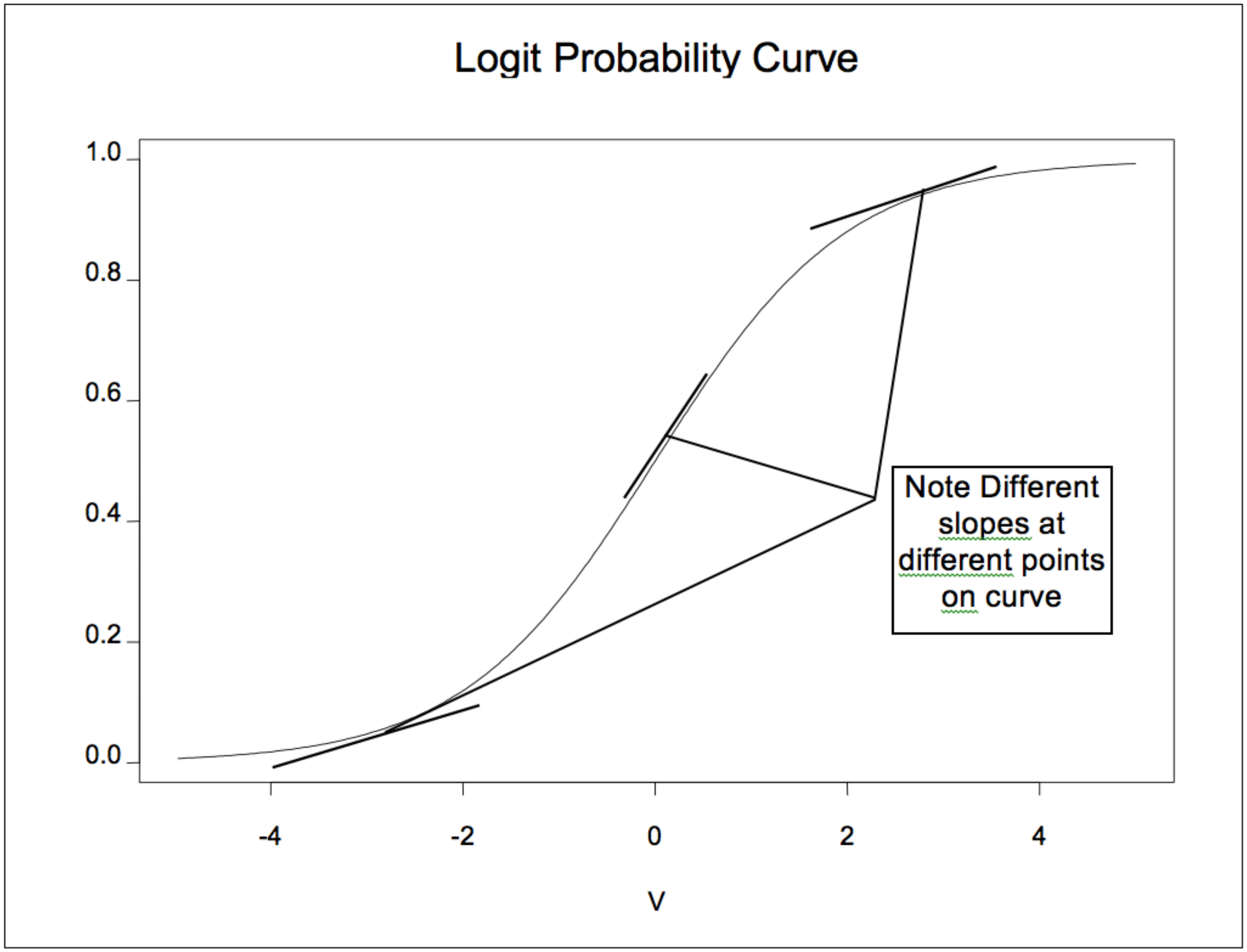
We can think of $V=\beta_0+\beta_1X_1+\ldots+\beta_kX_k$ as a "score." That is, what is the utility of buying.

As V gets large, the probability that Y=1 should get very close to 1. As V gets small, the probability that Y=1 should get close to zero.

$$\boxed{\Pr\left(Y=1\right)=\frac{\exp\left(V\right)}{1+\exp\left(V\right)}}$$

UCLA Anderson
SCHOOL of MANAGEMENT

c. The Logistic regression model



UCLAAnderson
SCHOOL of MANAGEMENT

d. Interpretation of Logistic slope coefficients

In a standard linear regression model, the slope coefficients should be interpreted as the average change in Y of a one unit change in the particular X variable.

We cannot interpret the slopes in a logit model as the change in the probability that Y=1 since the model is non-linear.

$$\boxed{\log\left(\frac{p}{1-p}\right)=\beta_0+\beta_1 x_1+\ldots+\beta_k x_k}$$

Log-odds is linear not the probability.

d. Interpretation of Logistic slope coefficients

The change in the probability of Y=1 with respect to a specific X variable is given by:

$$\frac{\partial Pr(Y=1|X)}{\partial X_j}=\beta_j Pr(Y=1|X)\big(1-Pr(Y=1|X)\big)$$

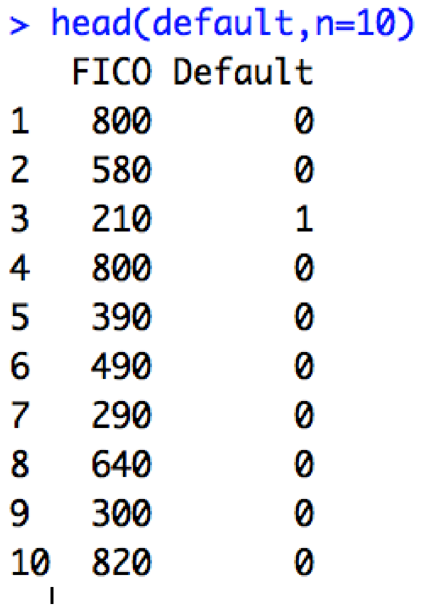
As a practical manner, we will simply use the fitted model to predict probabilities for different values of *X* and use this to determine change in probability for different values of *X*.

e. An Example

Consider the problem of predicting whether a borrower will default on a loan given their FICO score (300-850, higher is better) on application for the loan. We simulate some binary data (see code snippets for details).

Here X is the FICO score and Y = 1 if default, Y=0 if not.

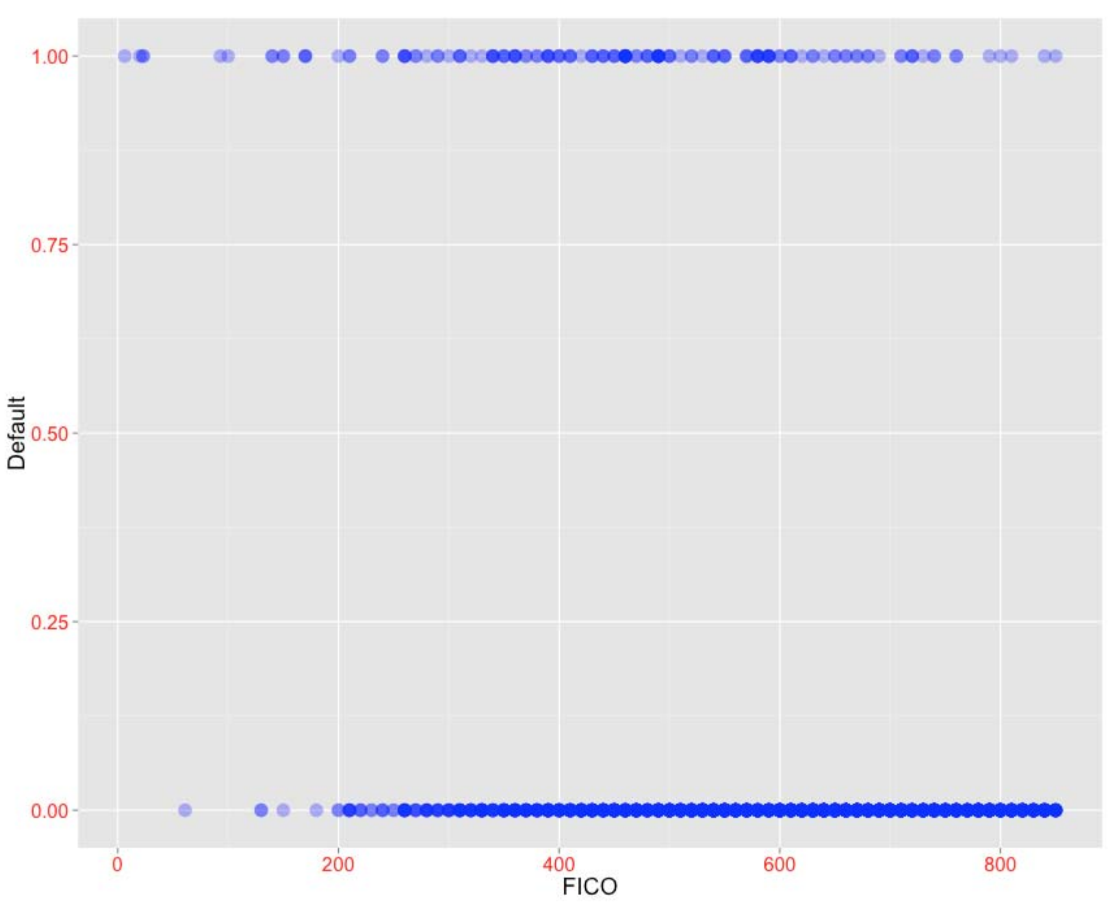
Let’s look at the data.



```
> head(default,n=10)
   FICO Default
1   800       0
2   580       0
3   210       1
4   800       0
5   390       0
6   490       0
7   290       0
8   640       0
9   300       0
10  820       0
```

e. An Example

If we attempt a scatterplot of y vs. x, we will only have two values of y. We use the alpha setting to see the density of X values.

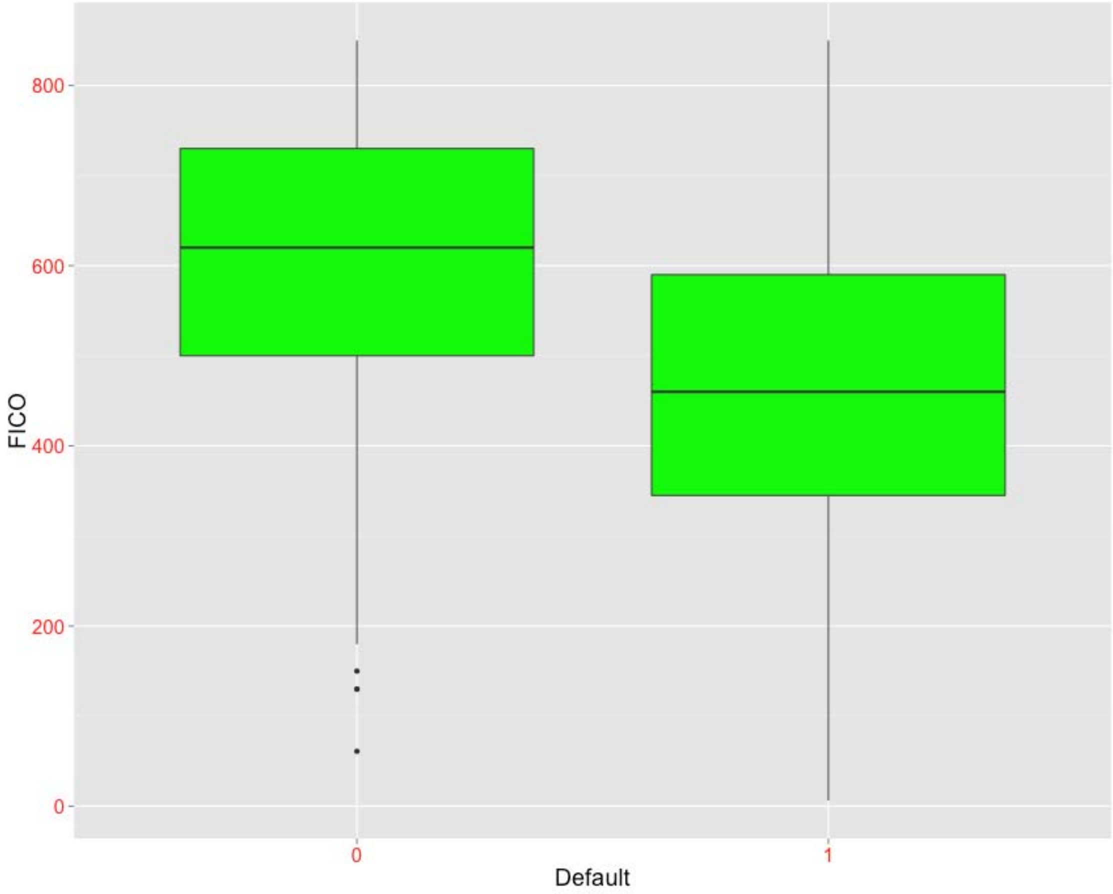


e. An Example

Hard to see what is going on. Let's do boxplots of FICO for the various values of Default.

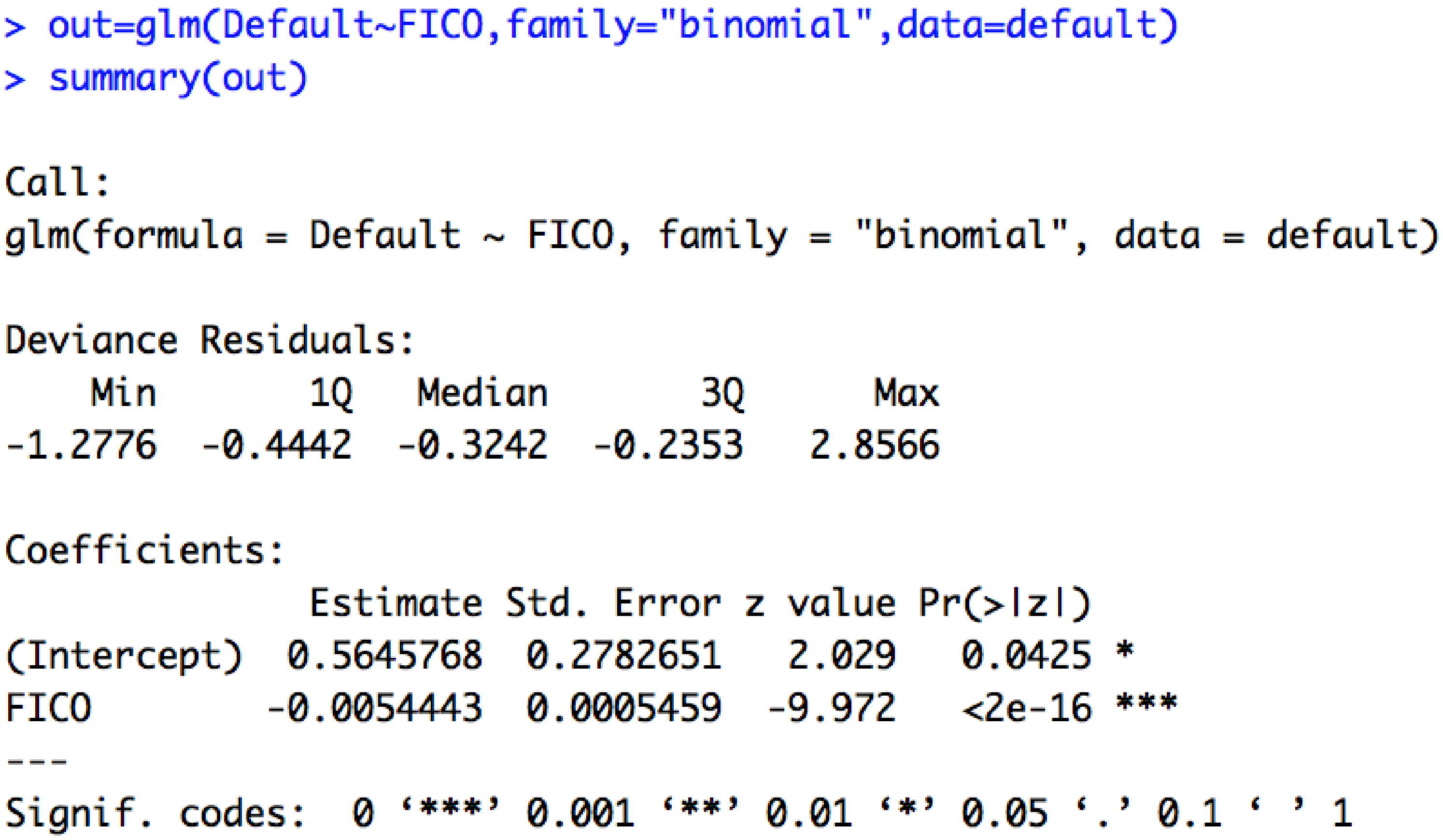
Can I use FICO to classify the observations?

Note that distributions of FICO scores overlap.



e. An Example

Let’s fit the model and show coefficients.



```
> out=glm(Default~FICO,family="binomial",data=default)
> summary(out)

Call:
glm(formula = Default ~ FICO, family = "binomial", data = default)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.2776  -0.4442  -0.3242  -0.2353   2.8566  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)  0.5645768  0.2782651   2.029   0.0425 *  
FICO        -0.0054443  0.0005459  -9.972   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

f. The Likelihood Function

How does this R fit this model to the data? There are no standard residuals. We can't do least squares.

The model is fit using the idea of maximum likelihood -- maximize the probability of observations.

Let's consider a coin toss of a not necessarily fair coin. Suppose we see 3 Heads in 10 coin tosses. Most of us would estimate the probability of a head for this coin to be 3/10.

Let's call θ the probability of a head. What is the likelihood of the data? It depends on theta!

f. The Likelihood Function

If we set $\theta = .5$, then what is the likelihood of the data?

$$L(\theta) = \theta^3(1-\theta)^{10-3}$$

$$if\ \theta = 0.5$$

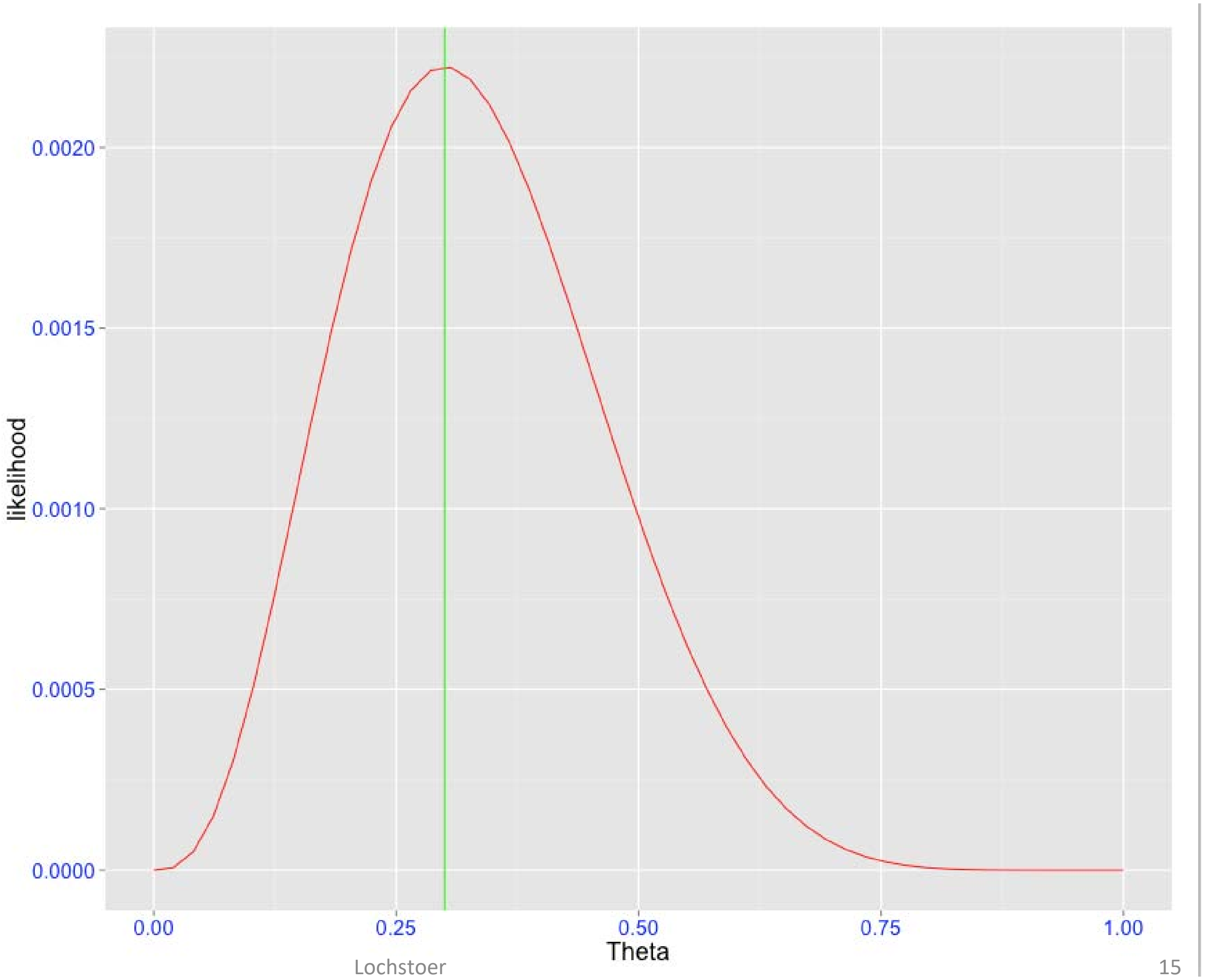
$$L(0.5) = 0.5^3(1-0.5)^7$$

Let's find the value of theta, which maximizes the "likelihood" of the observed data (3 Heads from 10).

Best way to do this is via a graph.

f. The Likelihood Function

Likelihood for coin toss, nhead=3, N=10



f. The Likelihood Function

So the idea is to pick logit regression parameters to maximize the probability the observed defaults given FICO.

For any value of beta0 and beta1, we can compute the probability of default from the Logit Model. The likelihood becomes the “score” for that pair of “guesses” of beta0 and beta1.

The likelihood is simply all of the probabilities for the observed defaults multiplied together. We ask the computer to maximize the likelihood for us by searching over possible values of beta0 and beta1.

f. The Likelihood Function

If we guess beta0 = 0 and beta1 = -.1,

$$Pr(Y_1) = Pr(No\ Default|FICO = 800)$$

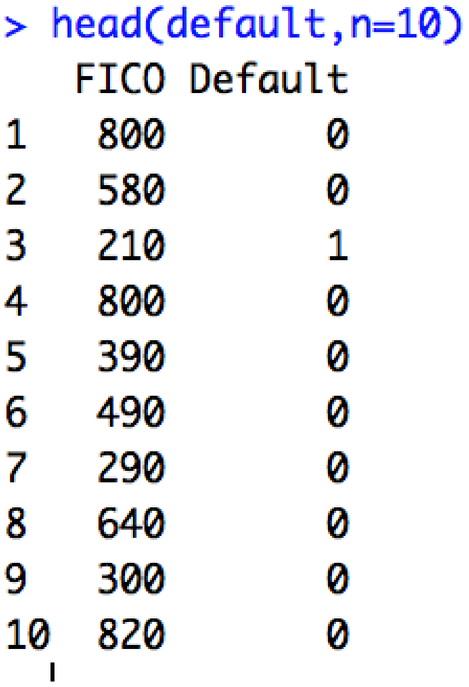
$$= 1 - \frac{exp(0 - 0.1 \times 800)}{1 + exp(0 - 0.1 \times 800)} = 1 - 0.018$$

$$Pr(Y_2) = Pr(No\ Default|FICO = 580)$$

$$= 1 - \frac{exp(0 - 0.1 \times 580)}{1 + exp(0 - 0.1 \times 580)} = 1 - 0.052$$

$$Pr(Y_3) = Pr(Default|FICO = 210)$$

$$= \frac{exp(0 - 0.1 \times 210)}{1 + exp(0 - 0.1 \times 210)} = 0.25$$



```
> head(default,n=10)
   FICO Default
1   800       0
2   580       0
3   210       1
4   800       0
5   390       0
6   490       0
7   290       0
8   640       0
9   300       0
10  820       0
```

f. The Likelihood Function

If we guess beta0 =0 and beta1 = -.1,

$$
\begin{aligned}
L(\beta_0 = 0, \beta_1 = -0.1) = Pr(Y_1|FICO_1) \times Pr(Y_2|FICO_2) \times \\
Pr(Y_3|FICO_3) \times \cdots \times Pr(Y_N|FICO_N) \\
= (1 - 0.18) \times (1 - 0.052) \times 0.25 \times \ldots
\end{aligned}
$$

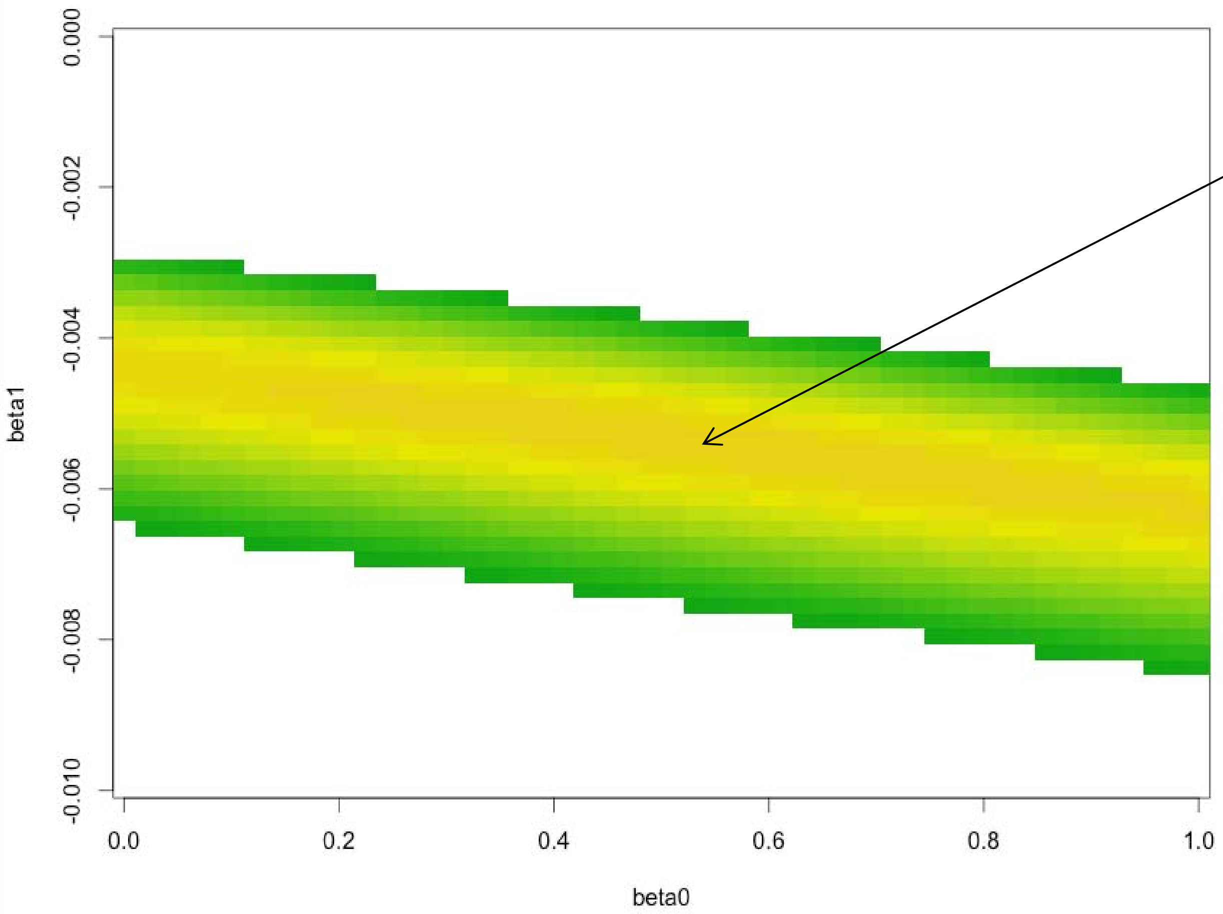
We ask the computer to find the values of the coefficients that maximize the likelihood of the observed data. Likelihood is the “scorecard” like SSE was for linear regression.

Using X values, we are trying to make the fitted probabilities of default as large as possible for all observations where Y=1 and as small as possible for all observations with Y=0.

This is called the method of Maximum Likelihood.

f. The Likelihood Function

Let's look at the surface that is being maximized:



Maximum Likelihood Estimate "top of the Mountain"

f. The Likelihood Function

Another way to see this is to write down the likelihood for the general logit model.

$$L(\beta \mid y, X) = \prod_{i=1}^{N} \Pr(Y_i = 1)^{y_i} \left(1 - \Pr(Y_i = 1)\right)^{1-y_i}$$

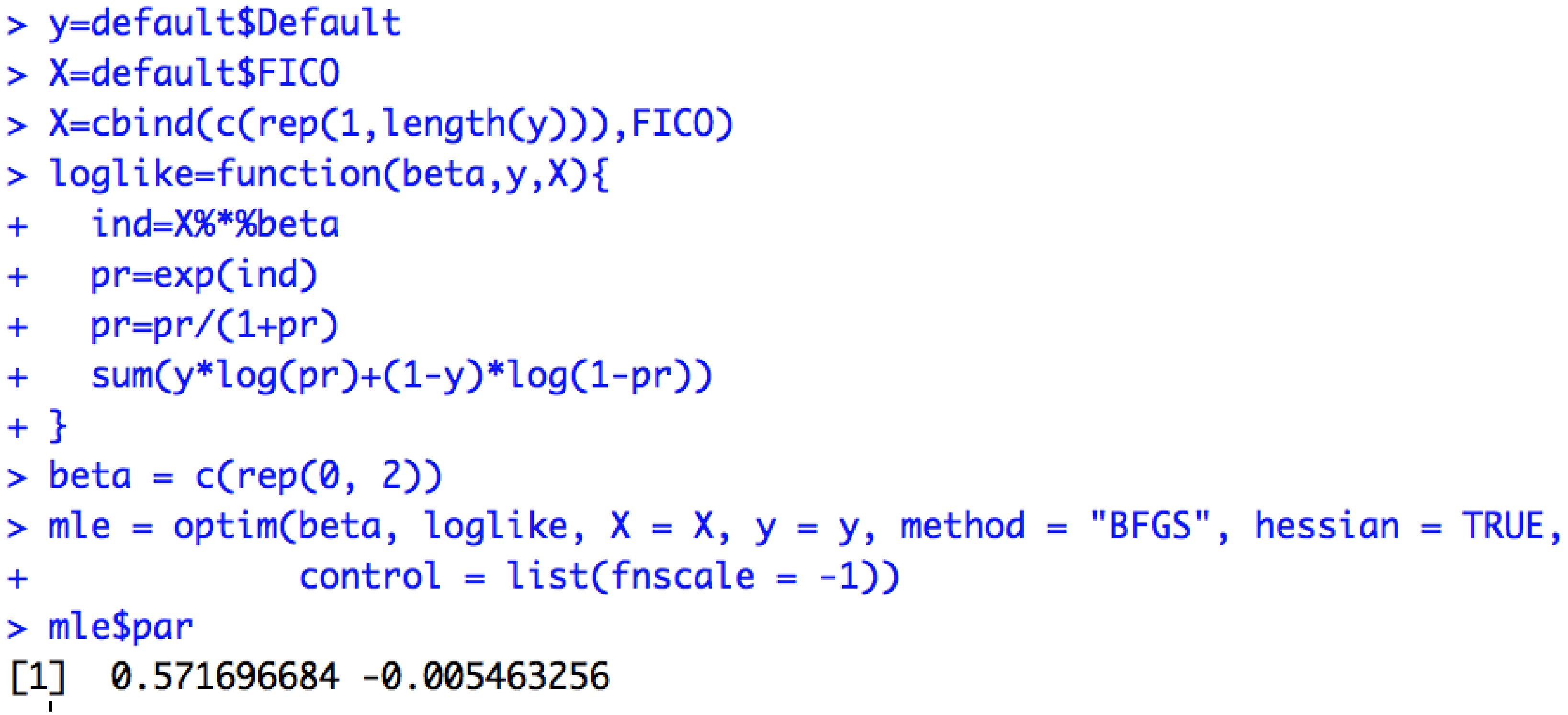
$$\Pr(Y_i = 1) = f(\beta) = \frac{\exp(x_i'\beta)}{1 + \exp(x_i'\beta)}$$

$$X = \begin{bmatrix} x_1' \\ \vdots \\ x_N' \end{bmatrix}$$

Let’s code it up and let R find the maximum!

f. The Likelihood Function

`optim()` is an optimizer that finds the “minimum” of any function you give it using numerical derivatives by default.

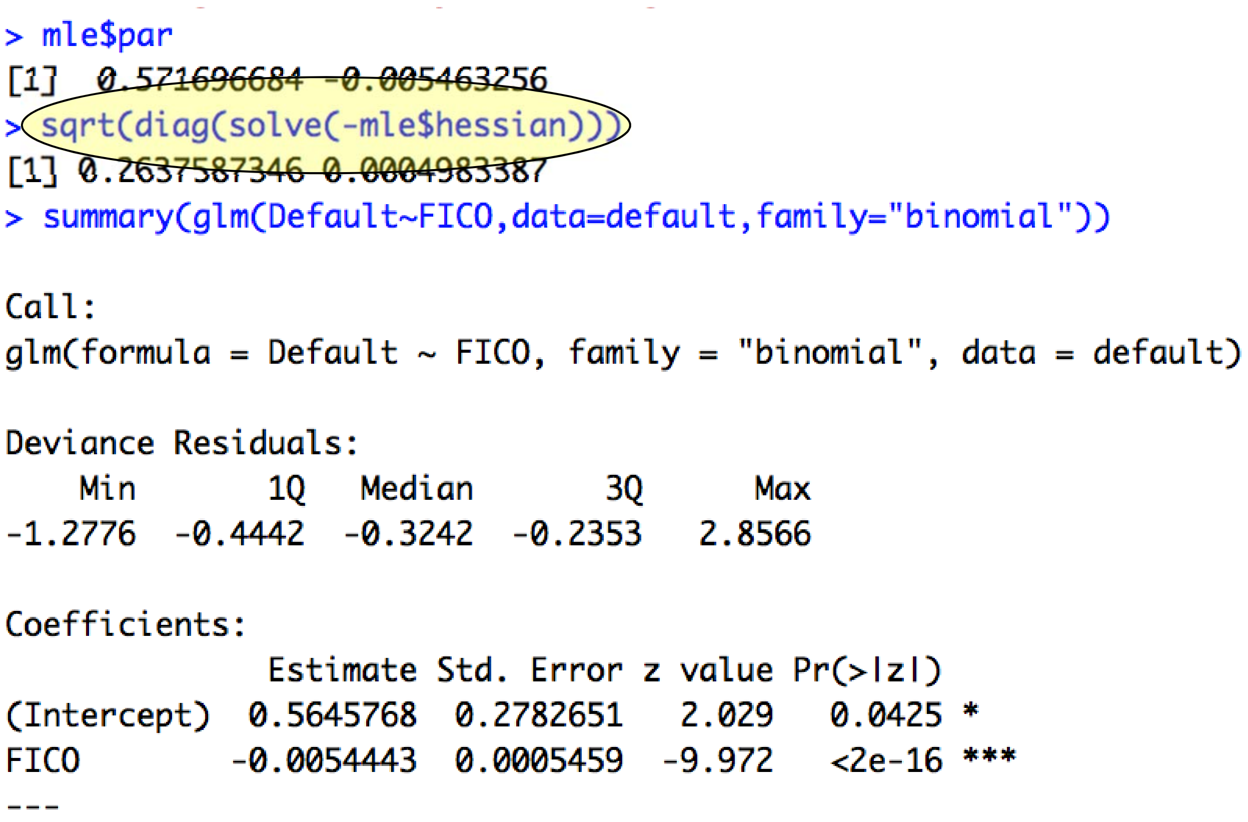


```
> y=default$Default
> X=default$FICO
> X=cbind(c(rep(1,length(y))),FICO)
> loglike=function(beta,y,X){
+   ind=X%*%beta
+   pr=exp(ind)
+   pr=pr/(1+pr)
+   sum(y*log(pr)+(1-y)*log(1-pr))
+ }
> beta = c(rep(0, 2))
> mle = optim(beta, loglike, X = X, y = y, method = "BFGS", hessian = TRUE,
+             control = list(fnscale = -1))
> mle$par
[1]  0.571696684 -0.005463256
```

UCLA Anderson
SCHOOL of MANAGEMENT

f. The Likelihood Function

`optim()` also returns the Hessian which is a measure of curvature that is also related to the standard errors for MLEs.



```
> mle$par
[1]  0.571696684 -0.005463256
> sqrt(diag(solve(-mle$hessian)))
[1] 0.2637587346 0.0004983387
> summary(glm(Default~FICO,data=default,family="binomial"))

Call:
glm(formula = Default ~ FICO, family = "binomial", data = default)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.2776  -0.4442  -0.3242  -0.2353   2.8566  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)  0.5645768  0.2782651   2.029   0.0425 *  
FICO        -0.0054443  0.0005459  -9.972   <2e-16 ***
---
```

f. Deviance

Note the "R^2" is not a natural measure of fit for a logistic regression.

- Errors are not normal (where variance makes sense) or even symmetric (so skewness is to be expected)
- Of this reason, we use "*Deviance*" as a measure of fit

Consider again the likelihood function:

$$L(\beta|y,X) = \prod_{i=1}^{N} Pr(Y_i = 1)^{y_i}\left(1 - Pr(Y_i = 1)\right)^{1-y_i}$$

Note that if there are enough parameters to fit each observation perfectly, we have that $L = 1$.

- Call this the *saturated model, M_s*

Let the *null* model, M_n, be the one with all coefficients, except the intercept coefficient, equals zero.

Let the proposed logit model be the candidate model, M_c, where the K betas are all estimated.

f. Deviance (cont’d)

Define the *null deviance* as:

$$d_{null} = 2(\ln L(M_s) - \ln L(M_n))$$

Define the *residual deviance* as:

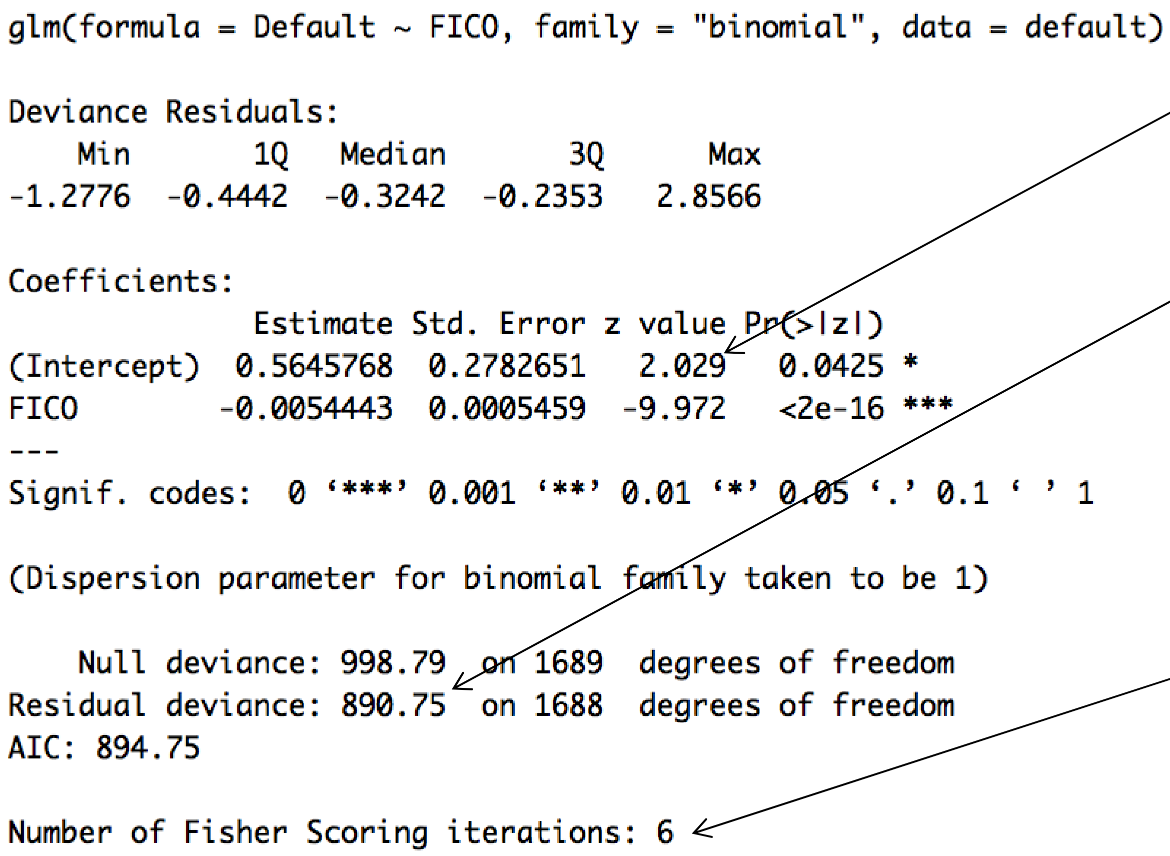
$$d_{residual} = 2(\ln L(M_s) - \ln L(M_c))$$

Notice that these are 2 times the difference in *log likelihood ratios* between the saturated and the null and candidate models

- Thus, from the standard ***likelihood ratio test***, this difference is Chi2-distributed with degrees of freedom equal to the number if observations in the sample minus the number of parameter in the non-saturated models

g. Simple Example continued

The printout shows us the results of this maximization:



```
glm(formula = Default ~ FICO, family = "binomial", data = default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.2776  -0.4442  -0.3242  -0.2353   2.8566

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.5645768  0.2782651   2.029   0.0425 *
FICO        -0.0054443  0.0005459  -9.972   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 998.79  on 1689  degrees of freedom
Residual deviance: 890.75  on 1688  degrees of freedom
AIC: 894.75

Number of Fisher Scoring iterations: 6
```

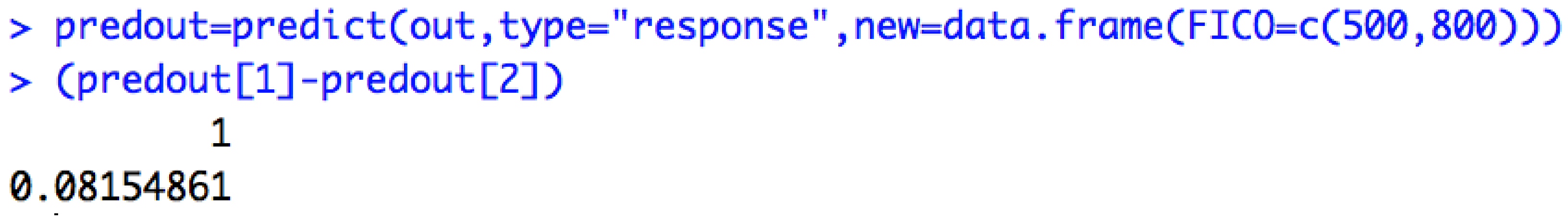
“t” tests

Residual Deviance is the analogue of SSE (smaller is better).

Took the computer 6 guesses!

g. A Simple Example continued

Let’s use the model to compute the expected change in default probability as we move FICO from 800 to 500.

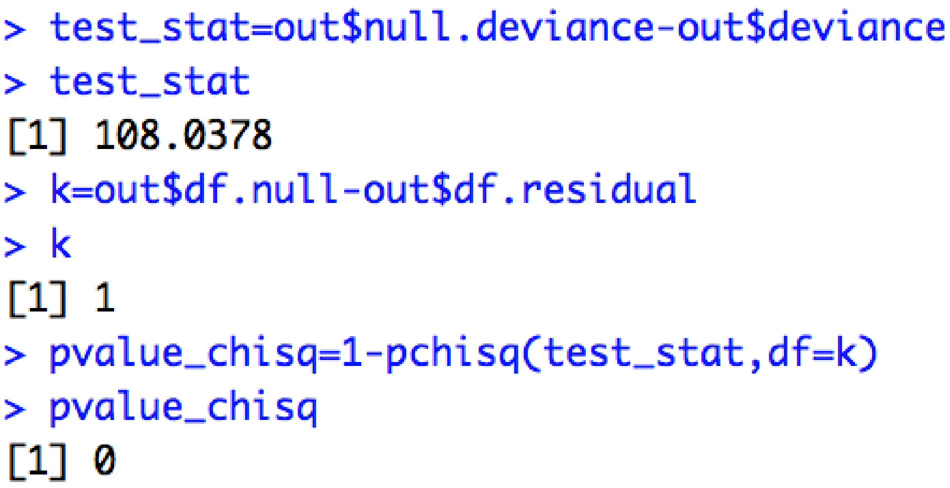


```
> predout=predict(out,type="response",new=data.frame(FICO=c(500,800)))
> (predout[1]-predout[2])
         1
0.08154861
```

g. A Simple Example continued

If you have only one regressor in the logistic regression, we just use the “t” (z) statistics to test for the significance of the regression. For more than one regressor, we need something like the overall F test. Here there is a chi-squared test that uses “deviance” computations.

Here the null is that all model slopes are zero. P-value can be used to test null. Deviance is sort of like SSE in a regular regression.



```
> test_stat=out$null.deviance-out$deviance
> test_stat
[1] 108.0378
> k=out$df.null-out$df.residual
> k
[1] 1
> pvalue_chisq=1-pchisq(test_stat,df=k)
> pvalue_chisq
[1] 0
```

g. A Simple Example continued

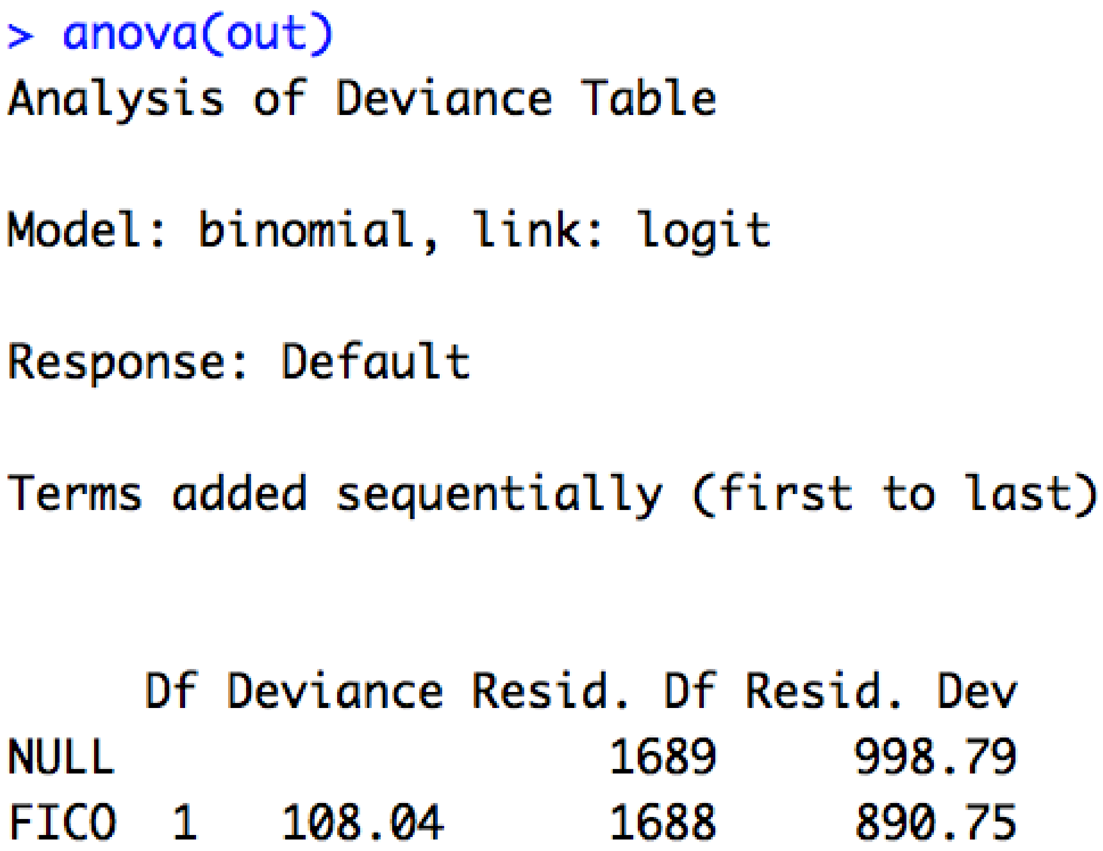
What is the null model here?

It is simply a logistic regression with an intercept term and no slope. This is a model for which the intercept will be chosen to make the fitted probabilities that Y = 1 equal the frequency for which Y = 1 in the data.

That is, the null model is simply to ignore X and compute the marginal probability that Y = 1 as opposed to the model which conditions on X!

g. A Simple Example continued

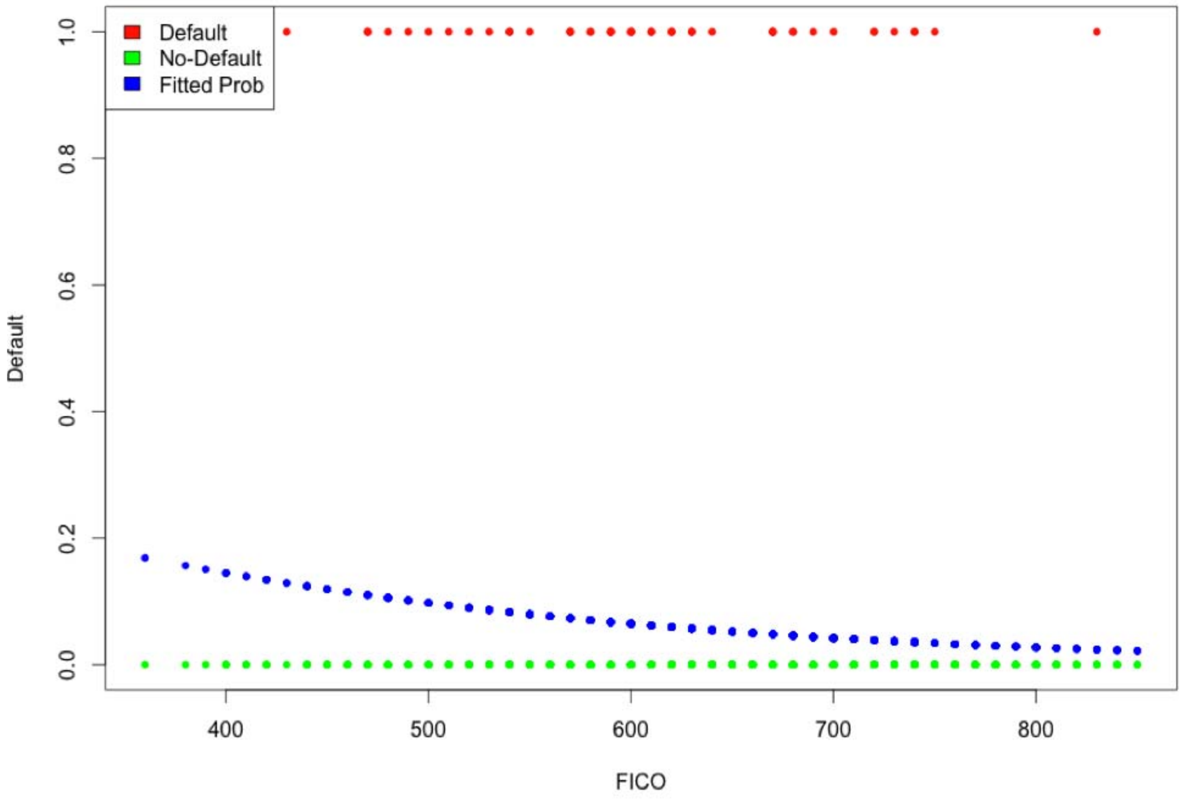
Alternatively, we could look at the `anova()` table for the model.



```
> anova(out)
Analysis of Deviance Table

Model: binomial, link: logit

Response: Default

Terms added sequentially (first to last)


     Df Deviance Resid. Df Resid. Dev
NULL                  1689     998.79
FICO  1   108.04      1688     890.75
```

g. A Simple Example continued

For logistic regression, there is no quantity like "*s*" or the prediction interval. The best we can do is compare outcomes to fitted probabilities in a plot. (see script)



h. A More Complicated Example

Consider a very common problem in Business Data Analytics:

Predict default on a consumer loan given information available at the time the loan is offered.

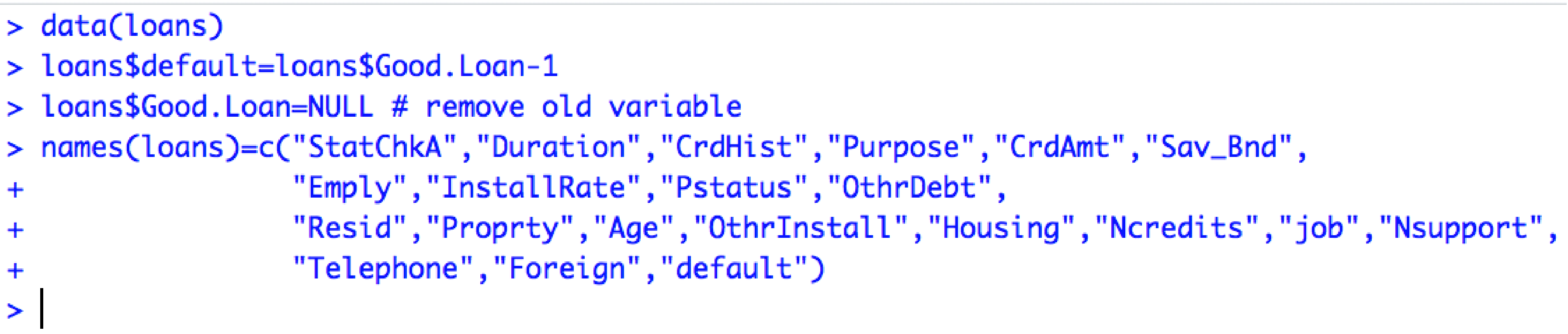
Explanatory Variables:

1. Credit History of Borrower
2. Terms of the Loan
3. Demographics (behavior usually trumps demos!)

h. A More Complicated Example

The `loans` dataset has information on 1,000 loans. Let’s build a logistic regression.

First, let’s compute default and rename variables:



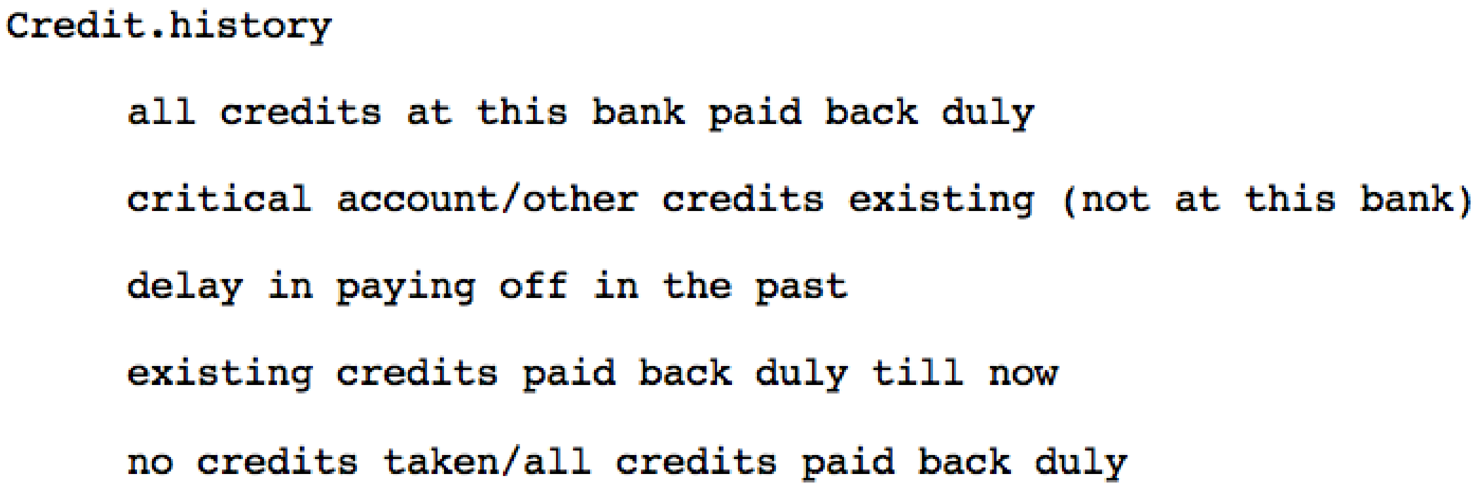
```
> data(loans)
> loans$default=loans$Good.Loan-1
> loans$Good.Loan=NULL # remove old variable
> names(loans)=c("StatChkA","Duration","CrdHist","Purpose","CrdAmt","Sav_Bnd",
+                "Emply","InstallRate","Pstatus","OthrDebt",
+                "Resid","Proprty","Age","OthrInstall","Housing","Ncredits","job","Nsupport",
+                "Telephone","Foreign","default")
> |
```

Note: we removed the “Good.Loan” variable from the copy of the dataset in our working environment!

h. A More Complicated Example

Most of the variables are categorical or qualitative variables.

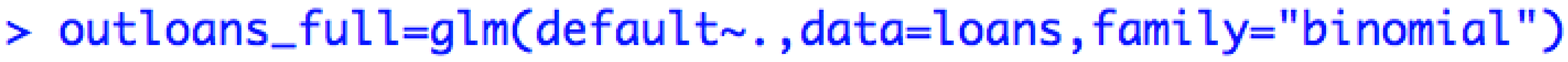
Examples:



```
Credit.history

     all credits at this bank paid back duly

     critical account/other credits existing (not at this bank)

     delay in paying off in the past

     existing credits paid back duly till now

     no credits taken/all credits paid back duly
```

h. A More Complicated Example

Let’s fit the full model. There will be a lot of coefficients, because R will create dummy variables for all of the categorical variables automatically.

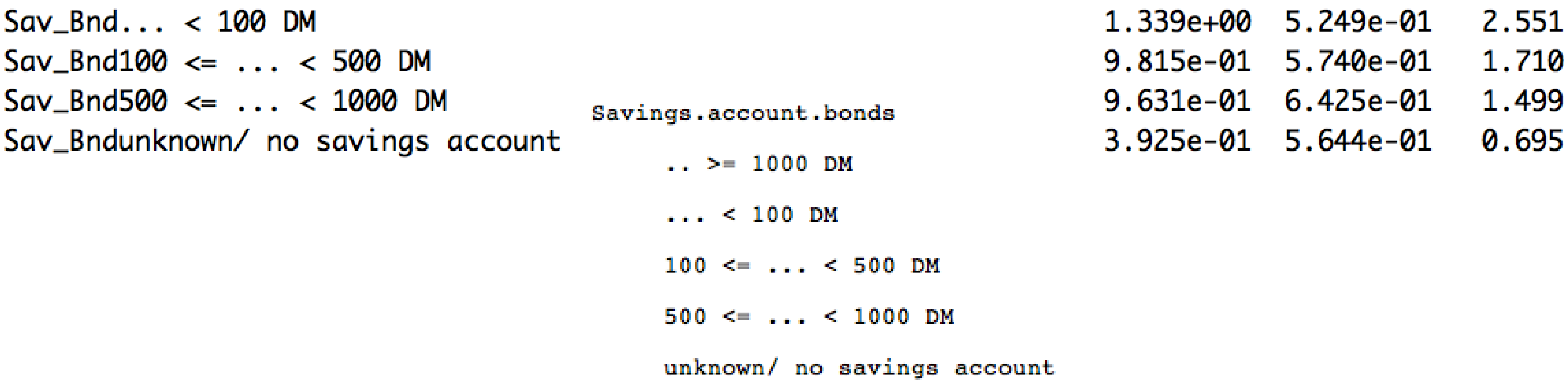


```
> outloans_full=glm(default~.,data=loans,family="binomial")
```

Remember the “.” in the formula means to regress default on all of the other variables in the dataset.

The “summary” of the model fit takes up a lot of space because of the long text descriptions of the factor levels.

h. A More Complicated Example



```
Sav_Bnd... < 100 DM                                    1.339e+00  5.249e-01   2.551
Sav_Bnd100 <= ... < 500 DM                             9.815e-01  5.740e-01   1.710
Sav_Bnd500 <= ... < 1000 DM                            9.631e-01  6.425e-01   1.499
Sav_Bndunknown/ no savings account                     3.925e-01  5.644e-01   0.695
```

```
Savings.account.bonds
    .. >= 1000 DM
    ... < 100 DM
    100 <= ... < 500 DM
    500 <= ... < 1000 DM
    unknown/ no savings account
```

Note: how R created dummy variables for each of the possible values of the `Sav_Bnd` variable except one (> 1000 in savings bonds)

You should interpret the coefficients as whether the probability of default will increase for the category versus the reference category.

e.g. if you have less than 100 in savings account, then you are more likely to default than someone with > 1000!

h. A More Complicated Example

```
                                                                    Estimate Std. Error z value Pr(>|z|)    
(Intercept)                                                       -2.967e+00  1.396e+00  -2.126 0.033543 *  
StatChkA... >= 200 DM / salary assignments for at least 1 year    -9.657e-01  3.692e-01  -2.616 0.008905 ** 
StatChkA0 <= ... < 200 DM                                         -3.749e-01  2.179e-01  -1.720 0.085400 .  
StatChkAno checking account                                       -1.712e+00  2.322e-01  -7.373 1.66e-13 ***
Duration                                                           2.786e-02  9.296e-03   2.997 0.002724 ** 
CrdHistcritical account/other credits existing (not at this bank) -1.579e+00  4.381e-01  -3.605 0.000312 ***
CrdHistdelay in paying off in the past                            -9.965e-01  4.703e-01  -2.119 0.034105 *  
CrdHistexisting credits paid back duly till now                   -7.295e-01  3.852e-01  -1.894 0.058238 .  
CrdHistno credits taken/all credits paid back duly                -1.434e-01  5.489e-01  -0.261 0.793921    
Purposecar (new)                                                   7.401e-01  3.339e-01   2.216 0.026668 *  
Purposecar (used)                                                 -9.264e-01  4.409e-01  -2.101 0.035645 *  
Purposedomestic appliances                                         2.173e-01  8.041e-01   0.270 0.786976    
Purposeeducation                                                   7.764e-01  4.660e-01   1.666 0.095718 .  
Purposefurniture/equipment                                        -5.152e-02  3.543e-01  -0.145 0.884391    
Purposeothers                                                     -7.487e-01  7.998e-01  -0.936 0.349202    
Purposeradio/television                                           -1.515e-01  3.370e-01  -0.450 0.653002    
Purposerepairs                                                     5.237e-01  5.933e-01   0.883 0.377428    
Purposeretraining                                                 -1.319e+00  1.233e+00  -1.070 0.284625    
CrdAmt                                                             1.283e-04  4.444e-05   2.887 0.003894 ** 
```

Cont’d on next page

h. A More Complicated Example

```
Sav_Bnd... < 100 DM                                         1.339e+00  5.249e-01   2.551 0.010729 *
Sav_Bnd100 <= ... < 500 DM                                  9.815e-01  5.740e-01   1.710 0.087293 .
Sav_Bnd500 <= ... < 1000 DM                                 9.631e-01  6.425e-01   1.499 0.133868
Sav_Bndunknown/ no savings account                          3.925e-01  5.644e-01   0.695 0.486765
Emply... < 1 year                                           2.097e-01  2.947e-01   0.712 0.476718
Emply1 <= ... < 4 years                                     9.379e-02  2.510e-01   0.374 0.708653
Emply4 <= ... < 7 years                                    -5.544e-01  3.007e-01  -1.844 0.065230 .
Emplyunemployed                                             2.766e-01  4.134e-01   0.669 0.503410
InstallRate                                                 3.301e-01  8.828e-02   3.739 0.000185 ***
Pstatusmale : divorced/separated                            2.755e-01  3.865e-01   0.713 0.476040
Pstatusmale : married/widowed                              -9.162e-02  3.118e-01  -0.294 0.768908
Pstatusmale : single                                       -5.406e-01  2.102e-01  -2.572 0.010113 *
OthrDebtguarantor                                          -1.415e+00  5.685e-01  -2.488 0.012834 *
OthrDebtnone                                               -4.360e-01  4.101e-01  -1.063 0.287700
Resid                                                       4.776e-03  8.641e-02   0.055 0.955920
Proprtyif not A121/A122 : car or other, not in attribute 6 -8.690e-02  2.313e-01  -0.376 0.707115
Proprtyreal estate                                         -2.814e-01  2.534e-01  -1.111 0.266630
Proprtyunknown / no property                                4.490e-01  4.130e-01   1.087 0.277005
Age                                                        -1.454e-02  9.222e-03  -1.576 0.114982
OthrInstallnone                                            -6.463e-01  2.391e-01  -2.703 0.006871 **
```

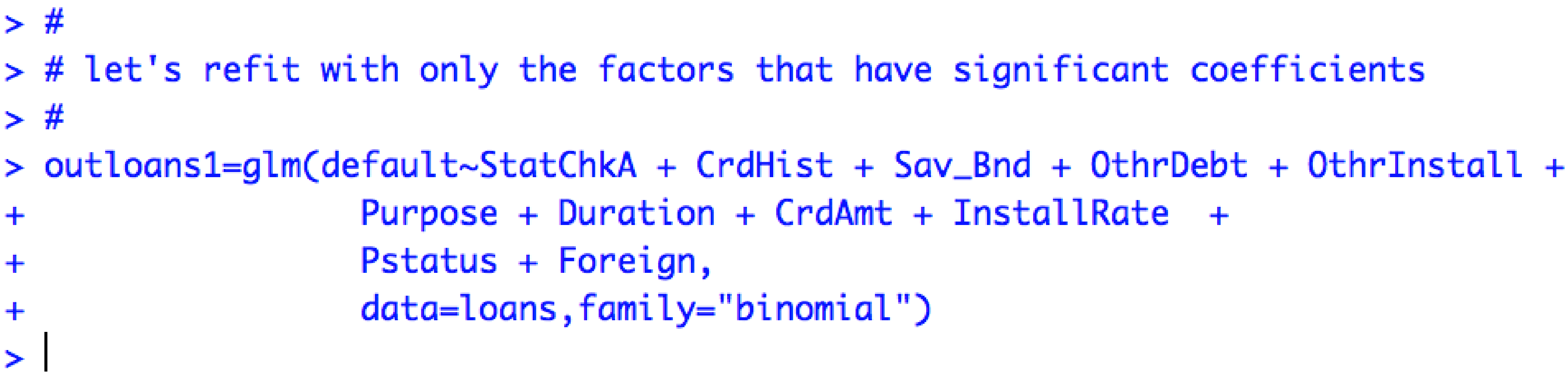
Cont’d on next page

h. A More Complicated Example

```
OthrInstallstores                                                 -1.232e-01  4.119e-01  -0.299 0.764878
Housingown                                                         2.402e-01  4.503e-01   0.534 0.593687
Housingrent                                                        6.839e-01  4.770e-01   1.434 0.151657
Ncredits                                                           2.721e-01  1.895e-01   1.436 0.151109
jobskilled employee / official                                     7.524e-02  2.845e-01   0.264 0.791419
jobunemployed/ unskilled - non-resident                           -4.795e-01  6.623e-01  -0.724 0.469086
jobunskilled - resident                                            5.666e-02  3.501e-01   0.162 0.871450
Nsupport                                                           2.647e-01  2.492e-01   1.062 0.288249
Telephoneyes, registered under the customers name                 -3.000e-01  2.013e-01  -1.491 0.136060
Foreignyes                                                         1.392e+00  6.258e-01   2.225 0.026095 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1221.73  on 999  degrees of freedom
Residual deviance:  895.82  on 951  degrees of freedom
AIC: 993.82

Number of Fisher Scoring iterations: 5
```

Take out the non-variables without any significance on a variable by variable or factor by factor basis. That is if ANY of the coefficients for each factor level are significant, keep it in!

h. A More Complicated Example



```
> #
> # let's refit with only the factors that have significant coefficients
> #
> outloans1=glm(default~StatChkA + CrdHist + Sav_Bnd + OthrDebt + OthrInstall +
+                     Purpose + Duration + CrdAmt + InstallRate  +
+                     Pstatus + Foreign,
+                     data=loans,family="binomial")
> |
```

Can we do a partial-f test to see if those factors I threw out should be kept out?

We don’t have a F-test but we do have a Chi-squared test based on the change in deviance or fit versus the number of variables removed. In other words, as we drop variables are we dropping degrees of freedom faster than reduction in fit?

h. A More Complicated Example

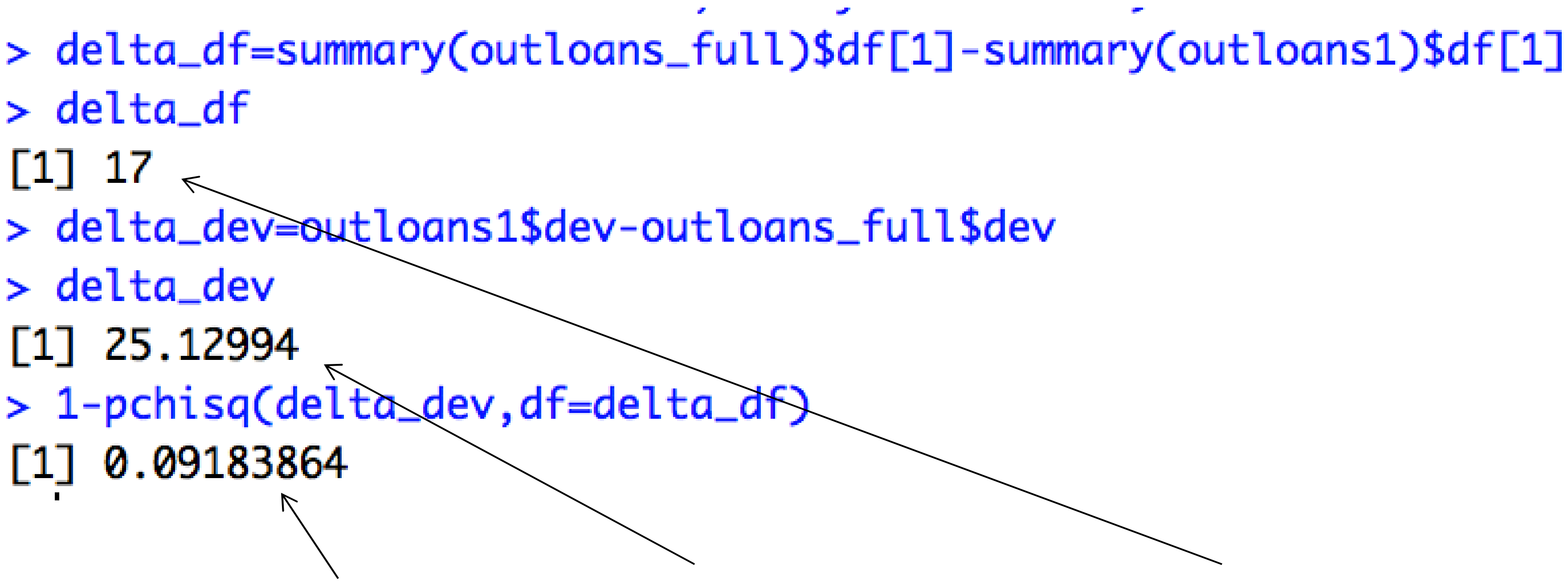
Inclusion-Exclusion Test:

Let’s compare the deviance from the full (all variables) with the restricted (insignificant variables are removed) just as we compared the R-squared of the full with the R-squared of the restricted for the F-test.

We also need to count how many variables were dropped. I can fetch this information from the `summary()` output.

UCLA Anderson
SCHOOL of MANAGEMENT

h. A More Complicated Example



```
> delta_df=summary(outloans_full)$df[1]-summary(outloans1)$df[1]
> delta_df
[1] 17
> delta_dev=outloans1$dev-outloans_full$dev
> delta_dev
[1] 25.12994
> 1-pchisq(delta_dev,df=delta_df)
[1] 0.09183864
```

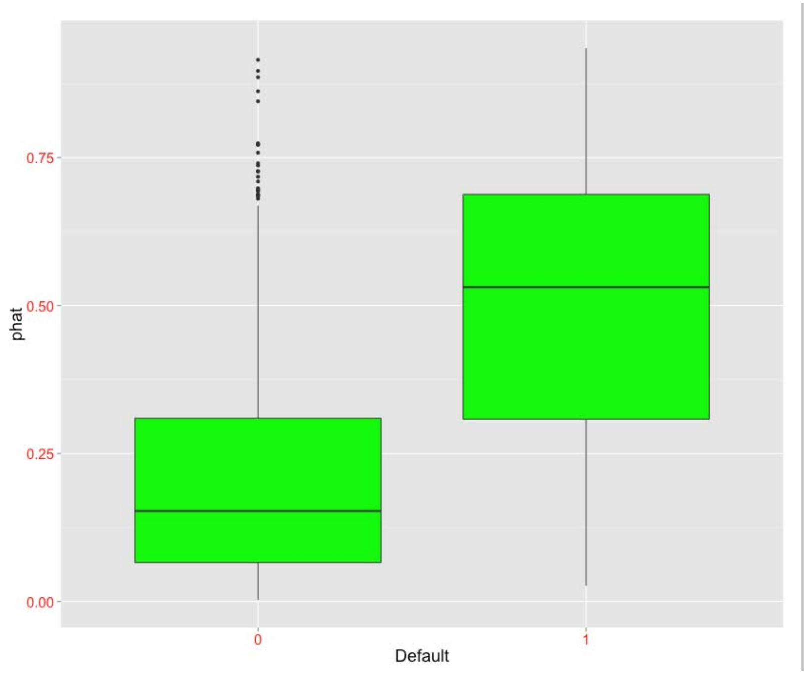
P-value for test. Null: variables have zero coefficients (should be thrown out).

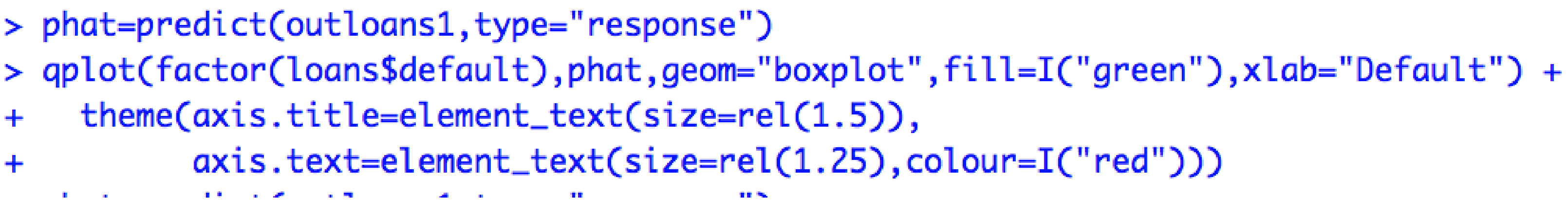
Change in fit. Fit is worse for simpler model by 25.13 “deviance” points.

17 variables thrown out (variables not factors)

h. A More Complicated Example

Let’s see how well the model does by plotting the distribution of fitted probabilities by default.





```
> phat=predict(outloans1,type="response")
> qplot(factor(loans$default),phat,geom="boxplot",fill=I("green"),xlab="Default") +
+   theme(axis.title=element_text(size=rel(1.5)),
+         axis.text=element_text(size=rel(1.25),colour=I("red")))
```

i. Lift Tables

Lift Table:

There is no “R-squared” for this model.

How should we evaluate the ability of the model to predict default?

A common practice is to create a “lift” table.

Sort the data by fitted probabilities and then compute the mean of the Y variable (mean response – in this case, mean default rate) for each decile of fitted probabilities.

If the model works well, then we should see much higher default rates for the higher fitted probabilities.

i. Lift Tables

This is converted in to a “lift” factor by dividing by the average response rate or overall average of Y.

A poor fitting model must not sort the observations well – the mean response rate for high fitted probabilities would only be marginally better than for small fitted probabilities.

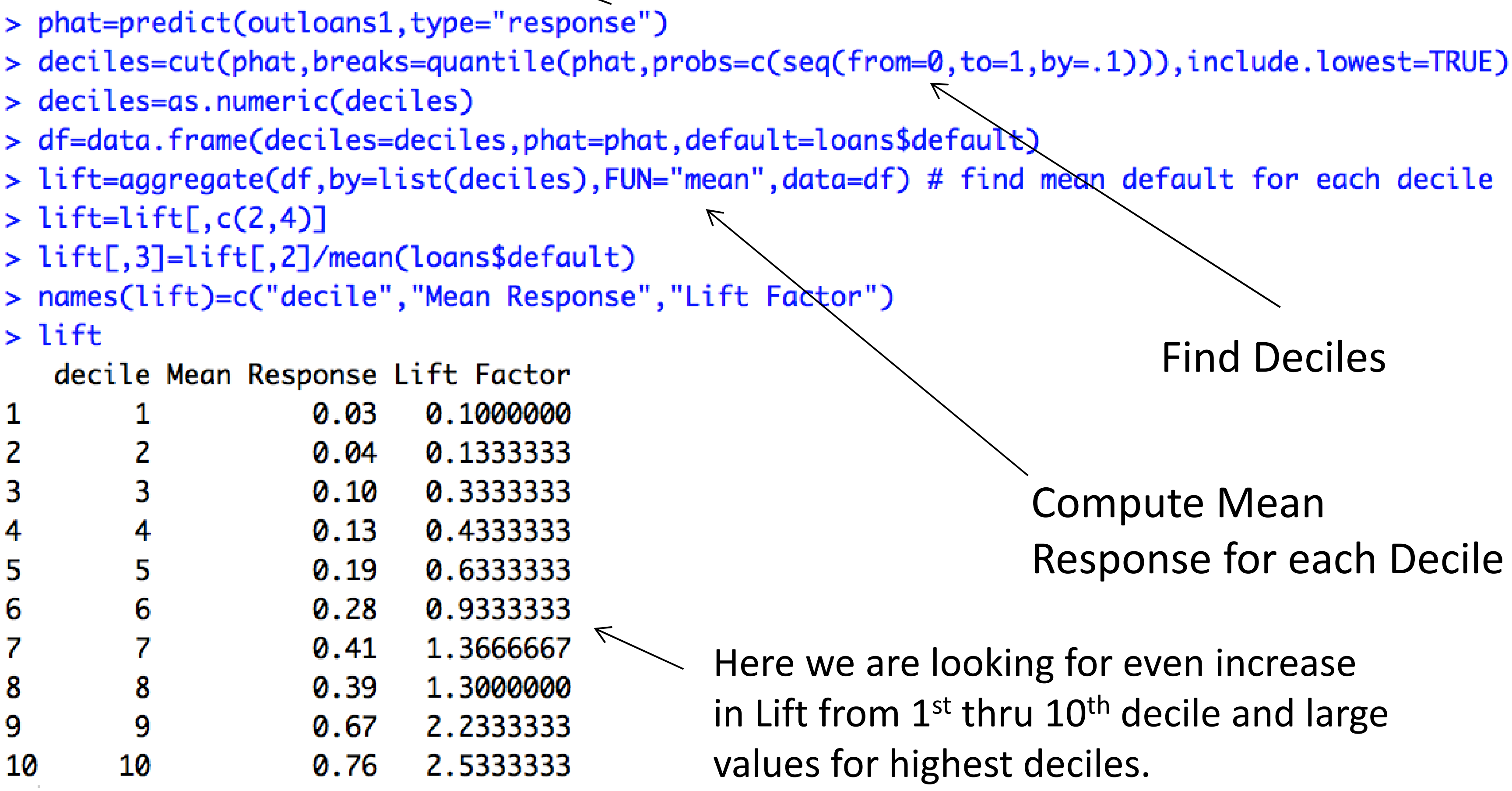
A good fit is evidenced by good discrimination of the data. The mean response rate for high fitted probabilities would be much greater than for low fitted probabilities.

This is called the “Lift Table”

UCLA Anderson
SCHOOL of MANAGEMENT

i. Lift Tables

Note that we need type="response" to get probs!



```
> phat=predict(outloans1,type="response")
> deciles=cut(phat,breaks=quantile(phat,probs=c(seq(from=0,to=1,by=.1))),include.lowest=TRUE)
> deciles=as.numeric(deciles)
> df=data.frame(deciles=deciles,phat=phat,default=loans$default)
> lift=aggregate(df,by=list(deciles),FUN="mean",data=df) # find mean default for each decile
> lift=lift[,c(2,4)]
> lift[,3]=lift[,2]/mean(loans$default)
> names(lift)=c("decile","Mean Response","Lift Factor")
> lift
   decile Mean Response Lift Factor
1       1          0.03   0.1000000
2       2          0.04   0.1333333
3       3          0.10   0.3333333
4       4          0.13   0.4333333
5       5          0.19   0.6333333
6       6          0.28   0.9333333
7       7          0.41   1.3666667
8       8          0.39   1.3000000
9       9          0.67   2.2333333
10     10          0.76   2.5333333
```

Find Deciles

Compute Mean Response for each Decile

Here we are looking for even increase in Lift from 1st thru 10th decile and large values for highest deciles.

j. ROC Curves

Receiver Operating Characteristics (ROC) graphs are useful for organizing classifiers and visualizing their performance

An alternative to Lift Tables (which was introduced instead of R2)

Popular metric so should know about this as well!

Benefits:

- Insensitive to changes in outcome distribution (overall positive outcomes (say, 1) versus negative outcomes (say, 0)
- This could be important if, say, instances of fraud changes from month to month
- Two-dimensional graph can be reduced to a single number of model fit (Area Under Curve) that has intuitive interpretation

j. Error Types

When considering whether to extend a loan or not, we can think of two types of borrowers

1. Good borrowers (repay loan)

2. Bad borrowers (default on loan)

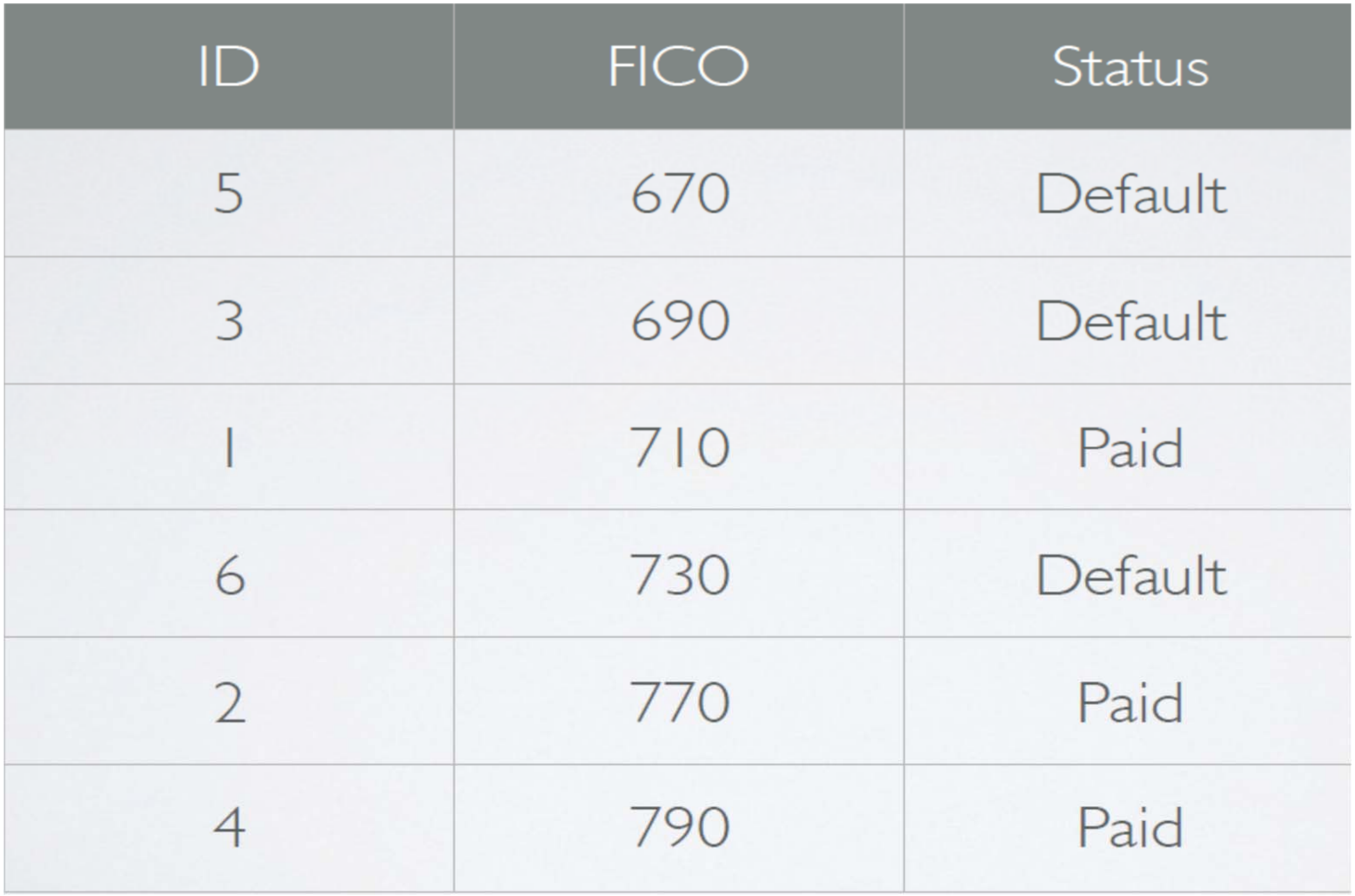
All lending institutions, including Market Place Lenders (MPLs) such as Lending Club, have sophisticated tools to try and distinguish between them

Is the goal to minimize defaults?

- No. Easy to achieve – do not extend loans
- Giving credit to bad borrowers is costly but denying credit to good borrowers is costly in terms of opportunity cost

Ideal model: Lend to 100% good borrowers, 0% bad borrowers

j. Error Types - An example



ID	FICO	Status
5	670	Default
3	690	Default
1	710	Paid
6	730	Default
2	770	Paid
4	790	Paid

FICO: Fair Isaac Co. A credit score.

- Assume this is the only information we have for this example

j. Error Types – FICO example

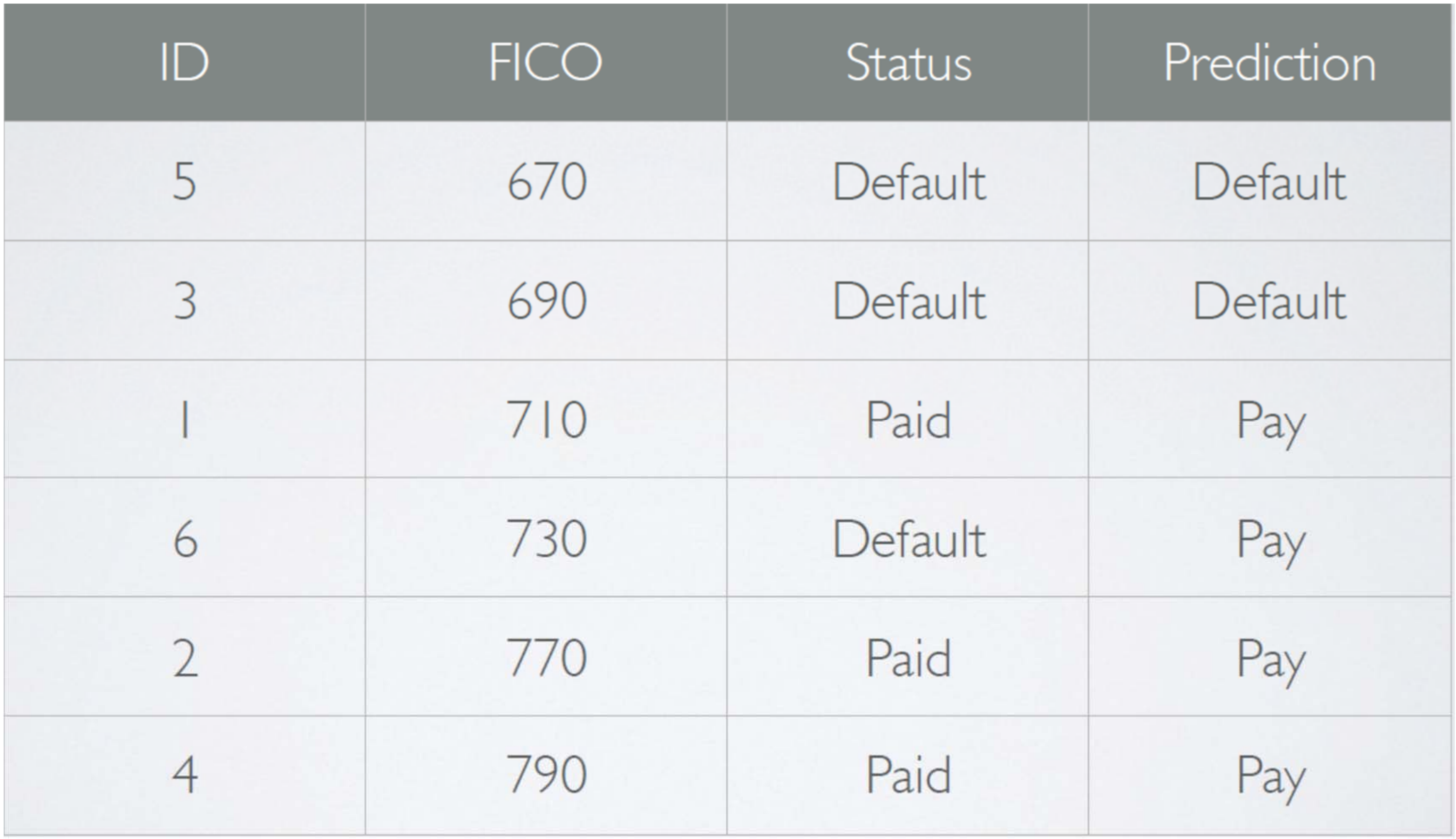
FICO is related to defaults, but not perfectly

- Other factors and randomness at play

You need to decide on the FICO cutoff below which you will deny credit

- Cutoff 1 = 700
- Cutoff 2 = 740

j. Using Cutoff of 700



ID	FICO	Status	Prediction
5	670	Default	Default
3	690	Default	Default
1	710	Paid	Pay
6	730	Default	Pay
2	770	Paid	Pay
4	790	Paid	Pay

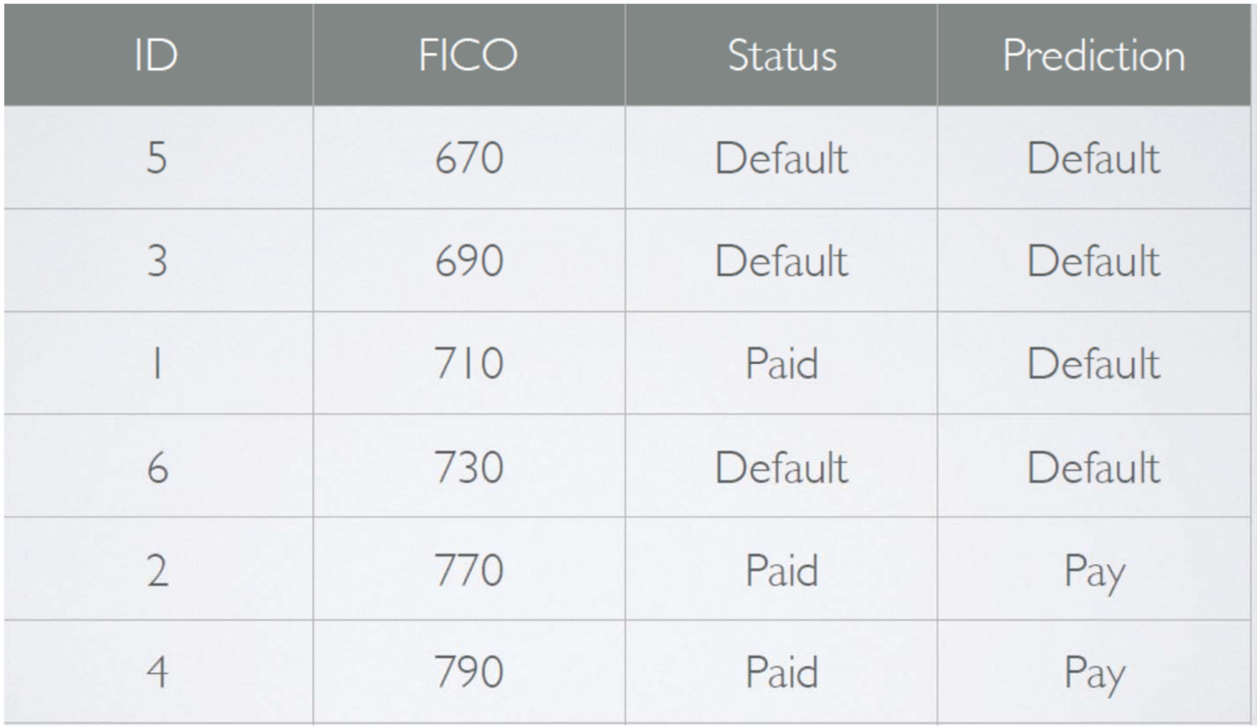
j. Model Performance @700

	True Paid	True Default
Predicted Paid	3	1
Predicted Default	0	2

In this case:

- True positive (TP) = 3
- False positive (FP) = 1

j. Using Cutoff @740



ID	FICO	Status	Prediction
5	670	Default	Default
3	690	Default	Default
I	710	Paid	Default
6	730	Default	Default
2	770	Paid	Pay
4	790	Paid	Pay

j. Model Performance @740

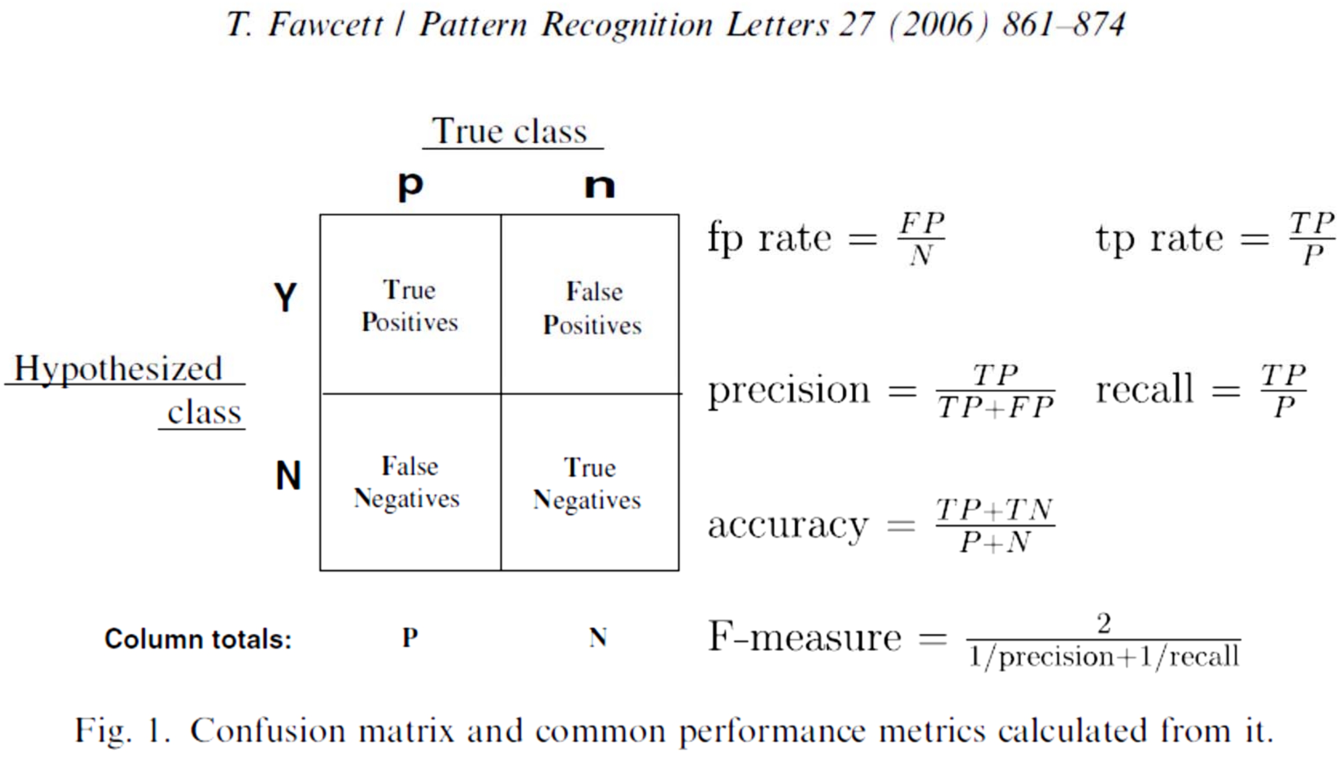
	True Paid	True Default
Predicted Paid	2	0
Predicted Default	1	3

In this case:

- True positive (TP) = 2
- False positive (FP) = 0

j. The Confusion Matrix

Classification problems can be represented by the aptly named *Confusion Matrix*



T. Fawcett / Pattern Recognition Letters 27 (2006) 861–874

		True class	
		p	n
Hypothesized class	Y	True Positives	False Positives
	N	False Negatives	True Negatives
Column totals:		P	N

$$\text{fp rate} = \frac{FP}{N} \qquad \text{tp rate} = \frac{TP}{P}$$

$$\text{precision} = \frac{TP}{TP+FP} \qquad \text{recall} = \frac{TP}{P}$$

$$\text{accuracy} = \frac{TP+TN}{P+N}$$

$$\text{F-measure} = \frac{2}{1/\text{precision}+1/\text{recall}}$$

Fig. 1. Confusion matrix and common performance metrics calculated from it.

j. The ROC Curve

Tracing out the true and false positives for different cutoffs of the score (in this case, FICO) gives us the data for the scatter plot that is the ***ROC Curve***

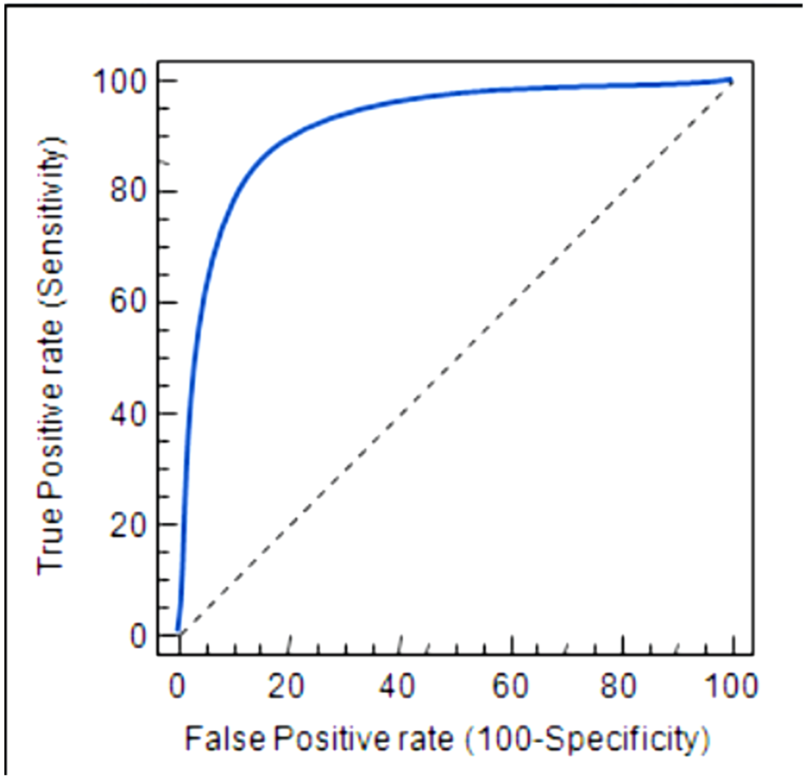
- Ideal model: vertical line at zero from zero to one (on y-axis), straight line at one thereafter from zero to one (on x-axis)

We use it to measure model performance

- The 45 degree line is the baseline, random guess case

Area Under Curve (AUC) summarizes model fit in one number

- Equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance.
- Random guess has probability 0.5, which is area under 45 degree line

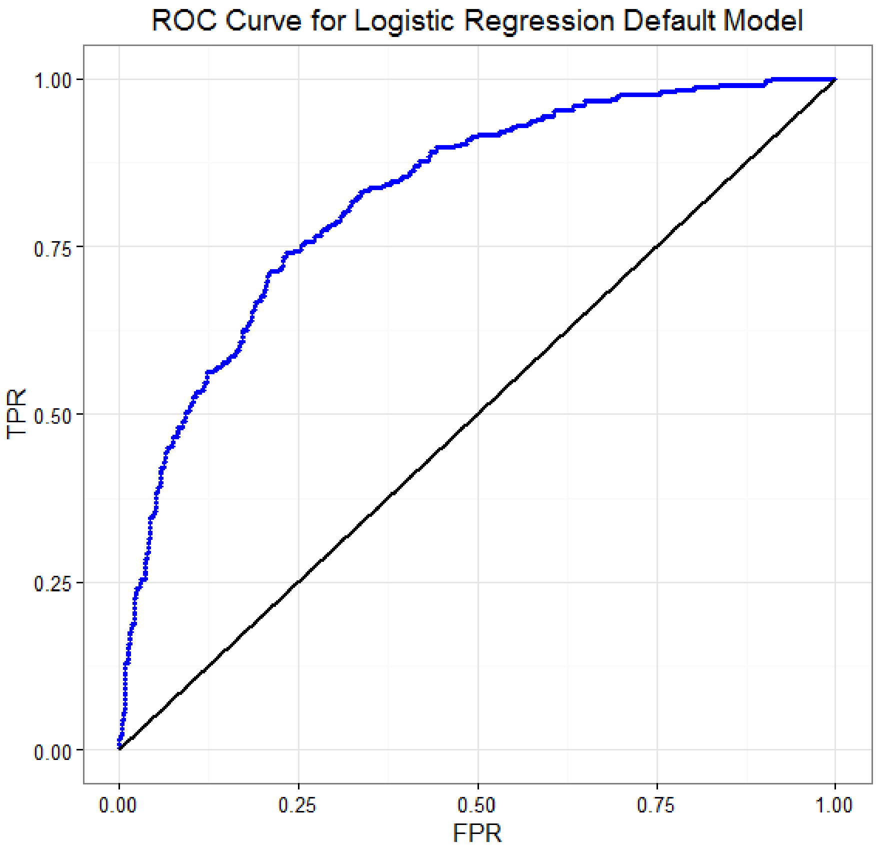


UCLAAnderson
SCHOOL of MANAGEMENT

j. ROC Curve from logistic regression model

Let’s construct the ROC curve from the restricted logistic regression model, ***where the “positive” in this case is the default prediction***

```
➢ simple_roc <- function(labels, scores)
➢ {
➢ labels <- labels[order(scores,
  decreasing=TRUE)]
➢ data.frame(TPR=cumsum(labels)/sum(labels),
  FPR=cumsum(!labels)/sum(!labels), labels)
➢ }
➢ glm_simple_roc <-
  simple_roc(loans$default==“1", phat)
➢ TPR <- glm_simple_roc$TPR
➢ FPR <- glm_simple_roc$FPR

➢ # plot the corresponding ROC curve
➢ q <-
  qplot(FPR,TPR,xlab="FPR",ylab="TPR",col=I("blue
  "),
   main="ROC Curve for Logistic Regression
   Default Model",size=I(0.75))
➢ # add straight 45 degree line from 0 to 1
➢ q + geom_segment(aes(x = 0, xend = 1, y = 0,
  yend = 1), size=I(1.0)) + theme_bw()
```

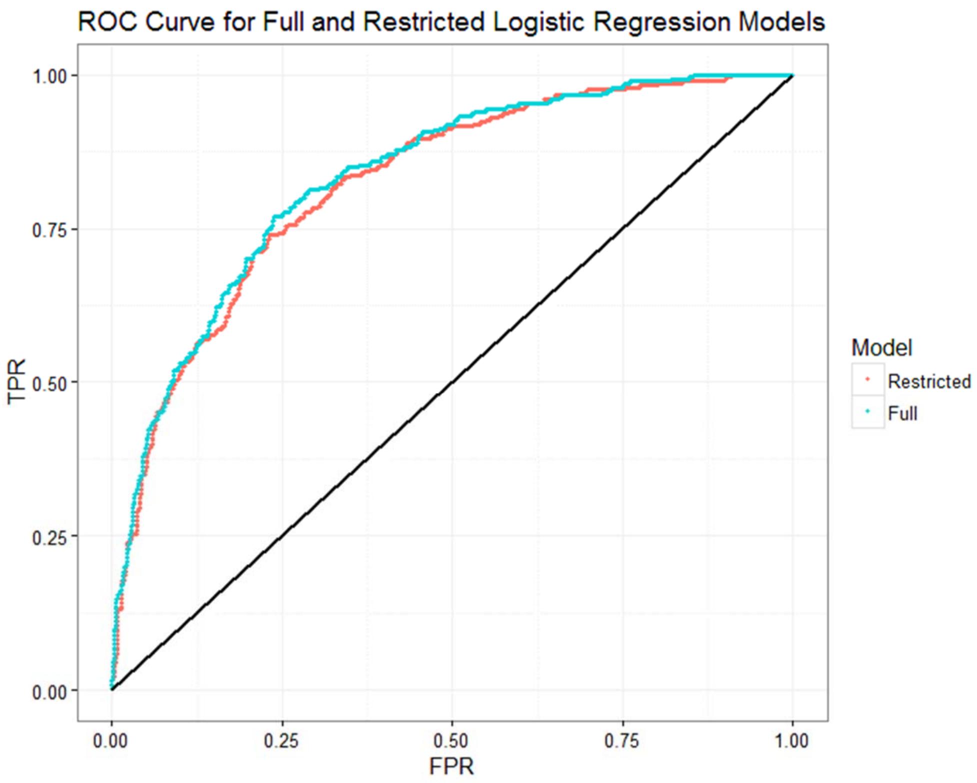


j. ROC Curve from logistic regression model

Let’s add the ROC for the full (unrestricted) logistic regression model

- Not much difference, as expected

```
➢ phat_full=predict(outloans_full,type="response
  ")
➢ glm_simple_roc_full <-
  simple_roc(loans$default=="1", phat_full)

➢ glm_simple_roc <- cbind(glm_simple_roc,Model =
  "Restricted")
➢ glm_simple_roc_full <-
  cbind(glm_simple_roc_full, Model = "Full")

➢  New_ROC <- rbind(glm_simple_roc,
  glm_simple_roc_full)

➢ q <- qplot(FPR,TPR,data = New_ROC, colour =
  Model, xlab="FPR",ylab="TPR",
   + main="ROC Curve for Full and Restricted
Logistic Regression Models",size=I(0.75))
> q + geom_segment(aes(x = 0, xend = 1, y = 0,
yend = 1), size=I(1.0), col = I("black")) +
theme_bw()
```



j. Bank profit maximization

Example:

Say you are a bank that wants to maximize profits from loans.

- Every time you give a loan and it is paid back you make money
- Every time you give a loan and borrower defaults you lose money

Problem: find FICO score (or probability of not defaulting from a more general model) ***cutoff*** for giving loan to an applicant that maximizes profits:

$$\max_{\{cutoff\}} LoansGiven(cutoff) \times ExpectedProfitPerLoan\,(cutoff)$$

UCLA Anderson
SCHOOL of MANAGEMENT

j. Bank profit maximization

With our default prediction model, the natural cutoff is to choose the highest accepted probability of default

- Note that from the bank's perspective, the positive outcome is no default, whereas the positive (high) outcome in our logistic regression was a default
  - I know. This is confusing. But, it's good to note that you have to be careful about these things when you are faced with this type of problem.
- This will often be how models are run/default prediction results are reported.
- Thus, a True Negative (**TN**) from our model is good (no default, loan given), whereas a False Negative (**FN**; default, loan given) is bad!
- Note: these are not *rates* but actual number of cases in the sample
  - You can divide by the number of observations in your sample if you like
  - The total number of loans you give does matter for your overall profit!
- Maximization problem can then be written:

$$\max_{\{cutoff\}} \mathrm{TN}(cutoff) \times Profit_{NoDefault} - \mathrm{FN}(cutoff) \times Loss_{Default}$$

j. Profit maximization and ROC curves

How is the ROC curve related to the profit maximization?

- *It is not that directly related*

1. We are not using TP or TN *rates* as the number of loans given matters for profits
2. We need profit per good outcome and loss per bad outcome as additional information in order to maximize

ROC curves are a measure of how informative a given model is relative to

A) A random guess for a given cutoff
B) Another candidate model (model horse race)

In addition, you can see ***where*** in the TPR versus FPR space the model performs well or not so well

j. Profit maximization and ROC curves

To see how the ROC curve relates to the profit maximization in our case, rewrite the profit maximization as follows

- *Note: N are total negative (no defaults), P are total positives (default) using the convention from our estimated logistic regression*
- *FP is number of false positives (not rate); TP is number of true positives*
- *FPR is false positive rate; TPR is true positive rate (as in ROC curve)*

$$\mathrm{TN} = \mathrm{N} - \mathrm{FP} = \mathrm{N}(1 - \mathrm{FPR})$$
$$\mathrm{FN} = \mathrm{P} - \mathrm{TP} = \mathrm{P}(1 - \mathrm{TPR})$$

So:

$$\max_{\{cutoff\}} \quad \begin{array}{l} \mathrm{N}\{1 - \mathrm{FPR}(cutoff)\} \times Profit_{NoDefault} \\ \quad \mathrm{P}\{1 - \mathrm{TPR}(cutoff)\} \times Loss_{Default} \end{array}$$

k. Lending Club

3
INVESTORS PROVIDE FUNDING
LendingClub
BORROWERS
MAKE MONTHLY PAYMENTS

In Problem Set 3, you will work with loan data from a large marketplace peer-to-peer lender: Lending Club

How does an online credit marketplace work?

Lending Club uses technology to operate a credit marketplace at a ***lower cost than traditional bank loan programs***, passing the savings on to borrowers in the form of lower rates and to investors in the form of solid returns. Borrowers who used a personal loan via Lending Club to consolidate debt or pay off high interest credit cards report in a survey that the interest rate on their loan was an average of 30% lower than they were paying on their outstanding debt or credit cards.

k. Lending Club

Lending Club is the world's largest marketplace connecting borrowers and investors, where consumers and small business owners lower the cost of their credit and enjoy a better experience than traditional bank lending, and investors earn attractive risk-adjusted returns.

Here's how it works:

- Customers interested in a loan complete a simple application at LendingClub.com
- Lending Club leverage online data and technology to quickly assess risk, determine a credit rating and assign appropriate interest rates. Qualified applicants receive offers in just minutes and can evaluate loan options with no impact to their credit score
- Investors ranging from individuals to institutions select loans in which to invest and can earn monthly returns
- The entire process is online, using technology to lower the cost of credit and pass the savings back in the form of lower rates for borrowers and solid returns for investors.

I. Propensity Scores

We have seen that we need to be very careful to control for variables in our regression analyses. Without randomized experimentation, we cannot interpret our regression models as prescriptive (or estimating causal effects) unless we control for relevant correlated variables.

What is a precise definition of a causal effect?

$Y_{i,1}$ is the response of person i to the treatment

$Y_{i,0}$ is the response of person i to the lack of treatment (control)

What is the causal effect for person i?

Causal Effect = $Y_{i,1} - Y_{i,0}$

Problem: We only observe one of these quantities for each person!

I. Propensity Scores

We either expose person i to the treatment (assign to experimental group) or not!

Example:
We create a YouTube video to promote a Nexus tablet. Some people watch the video and then we see whether they purchased the product and some did not.

> We don’t know what the folks who watched the video would have done had they not been exposed to the video.

and

> We don’t know what the folks who were not exposed to the video would have done had they been exposed to the video!

I. Propensity Scores

Another way of seeing this is

$$Y_i = \begin{cases} Y_{i,1} & \text{if assigned to treatment} \\ Y_{i,0} & \text{if assigned to control} \end{cases}$$

How do we estimate the treatment effect if we never observe the effect of the treatment for controls and the effect of the "null" or control treatment for experimentals?

What we would like to do is find for each person in the treatment group their "identical twin" in the control group. This is called a "matching" approach to estimating treatment effects. We would then simply average the difference in twin pairs.

UCLAAnderson
SCHOOL of MANAGEMENT

I. Propensity Scores

Suppose we can’t find good matches? Hard to do!

We then can use randomization to assign folks to treatment and control groups. Since randomization is not related to response to the treatment (or anything else), then, *on average*, there will be no difference between control and treatment groups except for the treatment effect.

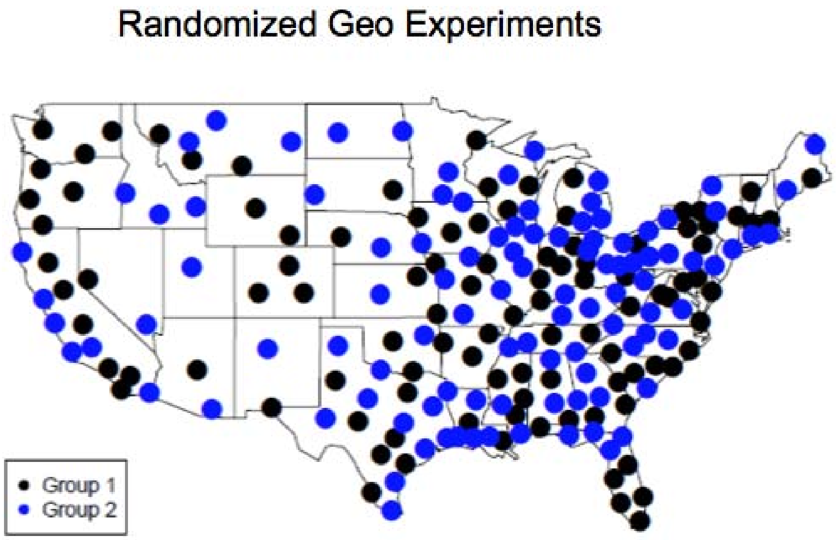
A simple treatment effect estimate is the difference in means between the treatment and control groups

$$\delta = \mu_1 - \mu_0 = E\left[Y_{i,1}\right] - E\left[Y_{i,0}\right]$$

$$\hat{\delta} = \bar{Y}_t - \bar{Y}_c$$

I. Propensity Scores

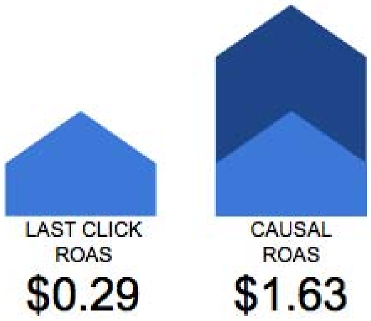
Below is the result of a random geographic assignment to an ad experiment condition.



The sales return on ad spend using "last click" attribution was $0.29 for every $1 in ad spend.

Using "causal attribution" it was $1.63 for every $1 of ad spend.

Causality makes a big difference!

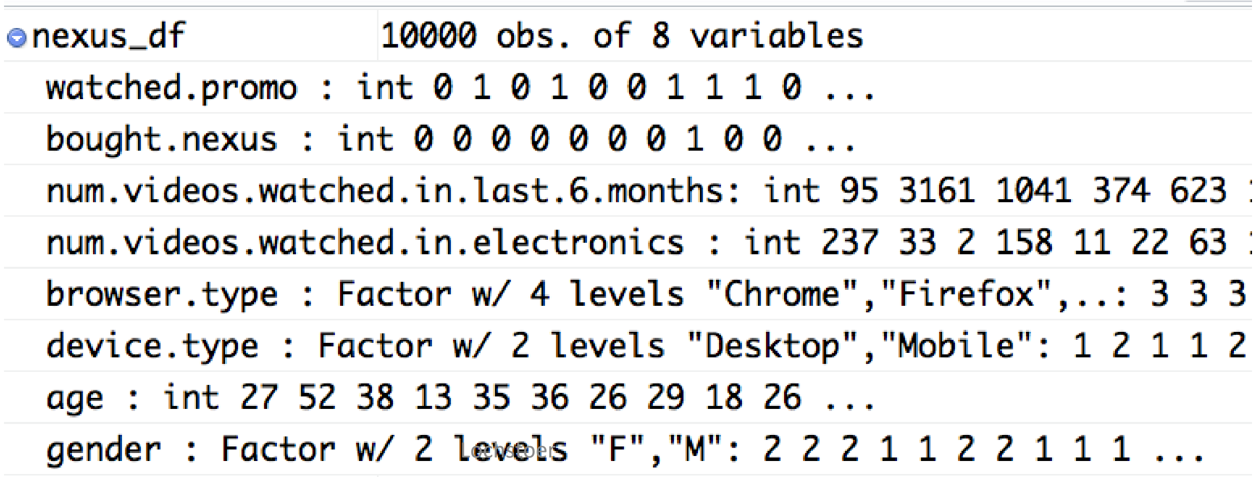


I. Propensity Scores

Suppose we have observational data and can’t implement random assignment to the treatment.

Back to the Nexus promo video example.
It is probable that the type of person who watches promo videos on tablets is probably much more likely to be interested and buy something than someone either at random or who didn’t watch the video.

Let’s read in the data and being to investigate.

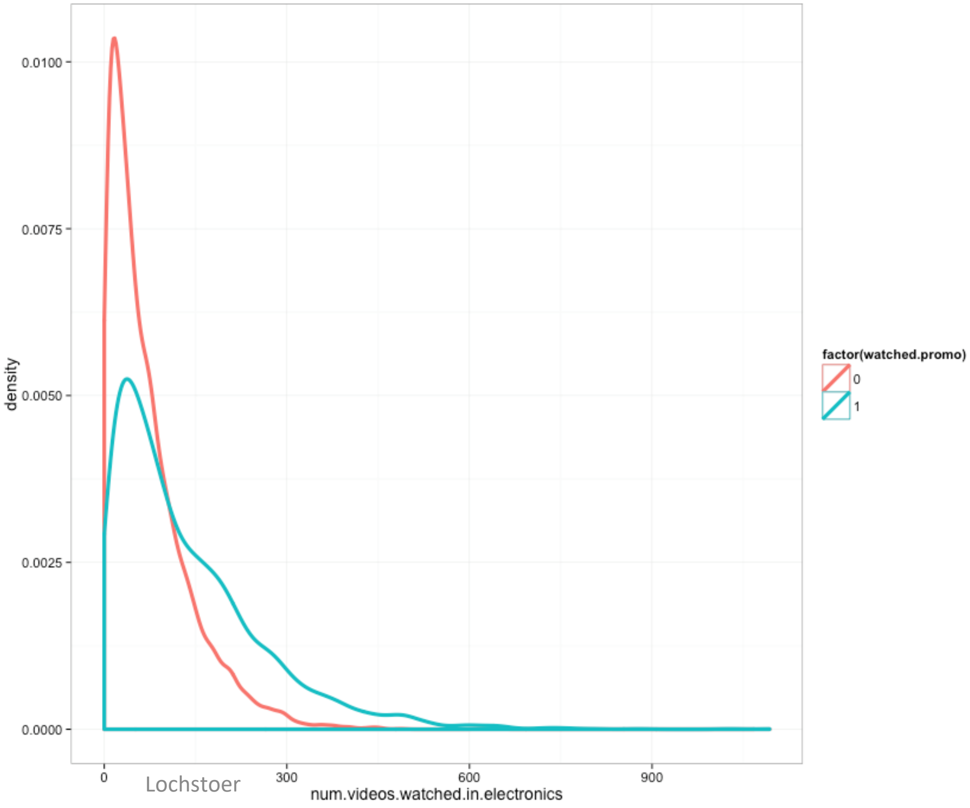


```
nexus_df            10000 obs. of 8 variables
  watched.promo : int 0 1 0 1 0 0 1 1 1 0 ...
  bought.nexus : int 0 0 0 0 0 0 0 1 0 0 ...
  num.videos.watched.in.last.6.months: int 95 3161 1041 374 623
  num.videos.watched.in.electronics : int 237 33 2 158 11 22 63
  browser.type : Factor w/ 4 levels "Chrome","Firefox",..: 3 3 3
  device.type : Factor w/ 2 levels "Desktop","Mobile": 1 2 1 1 2
  age : int 27 52 38 13 35 36 26 29 18 26 ...
  gender : Factor w/ 2 levels "F","M": 2 2 2 1 1 2 2 1 1 1 ...
```

I. Propensity Scores

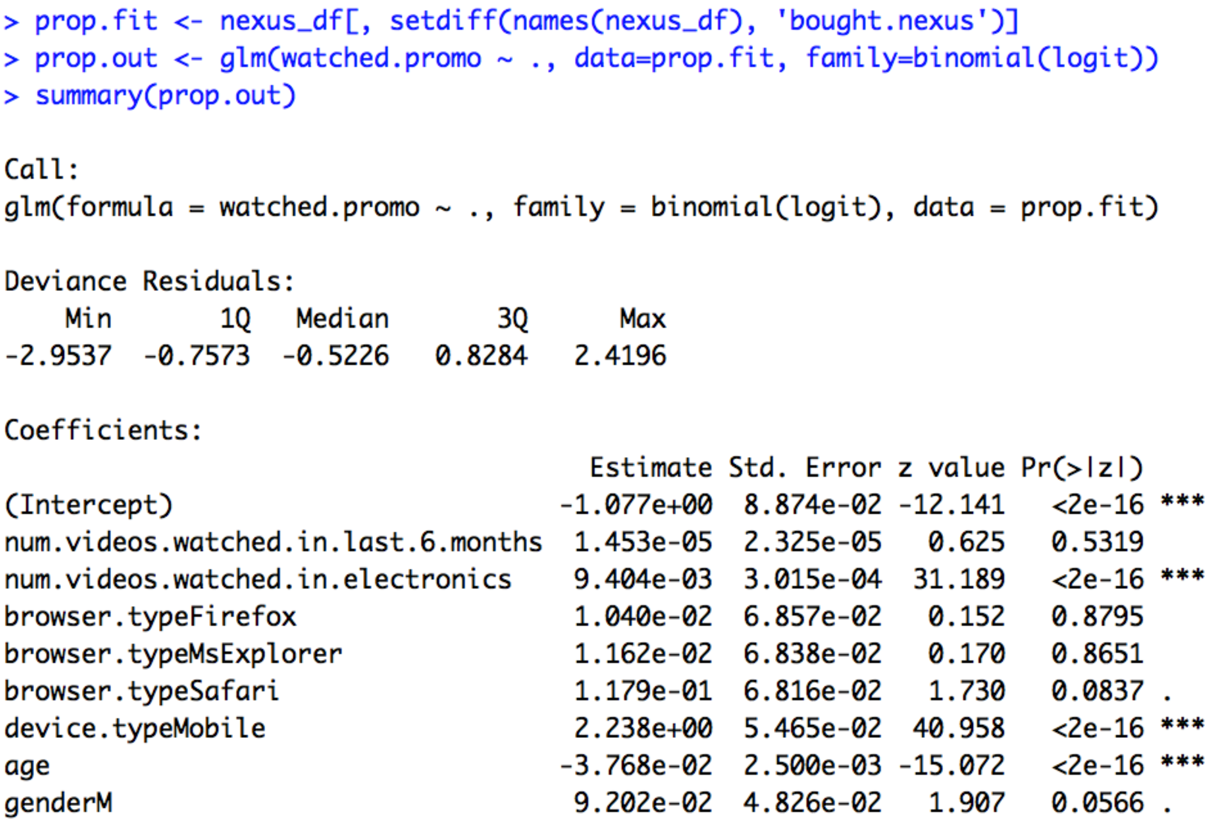
We have about 5000 folks who watched the promo video with a “random” sample of others who did not.

Probably those who watched the video were NOT a random sample. True, that!



I. Propensity Scores

Let’s see how well we can predict who watched the promo video on the basis of the variables we have. Confirms that we don’t have a random sample.



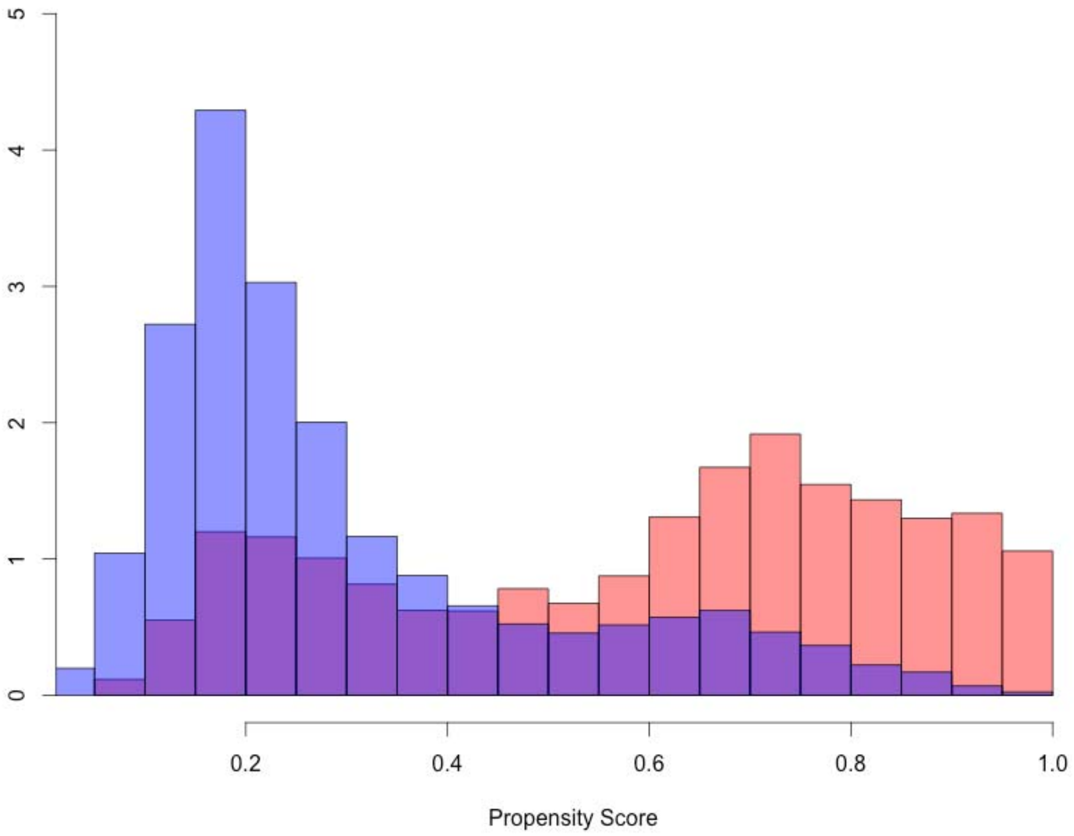
```
> prop.fit <- nexus_df[, setdiff(names(nexus_df), 'bought.nexus')]
> prop.out <- glm(watched.promo ~ ., data=prop.fit, family=binomial(logit))
> summary(prop.out)

Call:
glm(formula = watched.promo ~ ., family = binomial(logit), data = prop.fit)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9537  -0.7573  -0.5226   0.8284   2.4196

Coefficients:
                                  Estimate Std. Error z value Pr(>|z|)
(Intercept)                     -1.077e+00  8.874e-02 -12.141   <2e-16 ***
num.videos.watched.in.last.6.months  1.453e-05  2.325e-05   0.625   0.5319
num.videos.watched.in.electronics    9.404e-03  3.015e-04  31.189   <2e-16 ***
browser.typeFirefox              1.040e-02  6.857e-02   0.152   0.8795
browser.typeMsExplorer           1.162e-02  6.838e-02   0.170   0.8651
browser.typeSafari               1.179e-01  6.816e-02   1.730   0.0837 .
device.typeMobile                2.238e+00  5.465e-02  40.958   <2e-16 ***
age                             -3.768e-02  2.500e-03 -15.072   <2e-16 ***
genderM                          9.202e-02  4.826e-02   1.907   0.0566 .
```

UCLA Anderson
SCHOOL of MANAGEMENT

I. Propensity Scores

Let’s look at the distribution of propensity scores for those who did watch video and those who did not. Large differences!

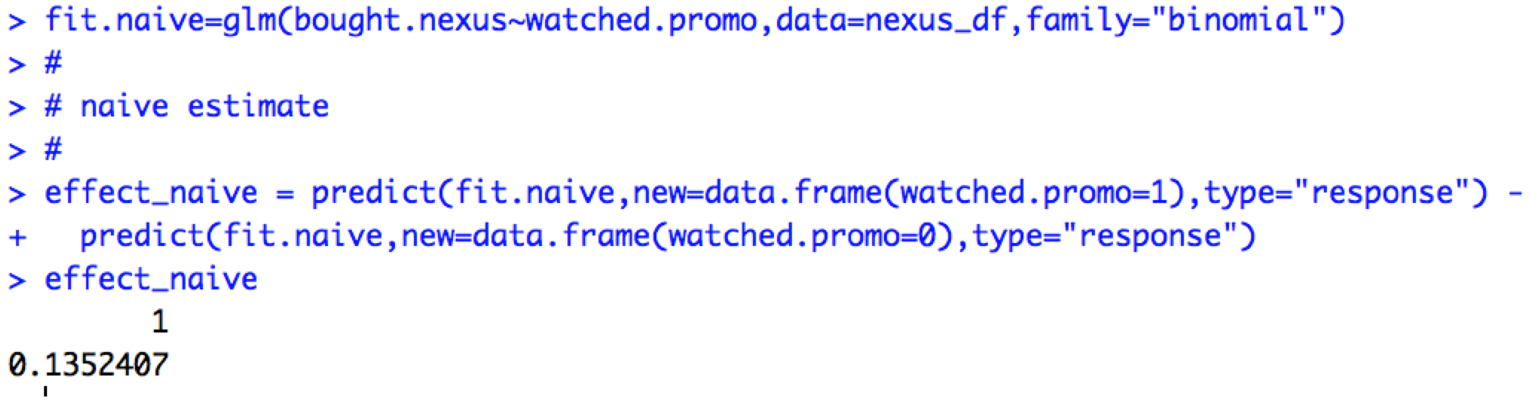
Did not Watch Promo



Watched Promo

I. Propensity Scores

Will this matter in estimating the effect? Suppose we just regard the two groups as randomly selected. Then we would simply compare the probability of purchase across treatment and control.



```
> fit.naive=glm(bought.nexus~watched.promo,data=nexus_df,family="binomial")
> #
> # naive estimate
> #
> effect_naive = predict(fit.naive,new=data.frame(watched.promo=1),type="response") -
+   predict(fit.naive,new=data.frame(watched.promo=0),type="response")
> effect_naive
        1 
0.1352407 
```

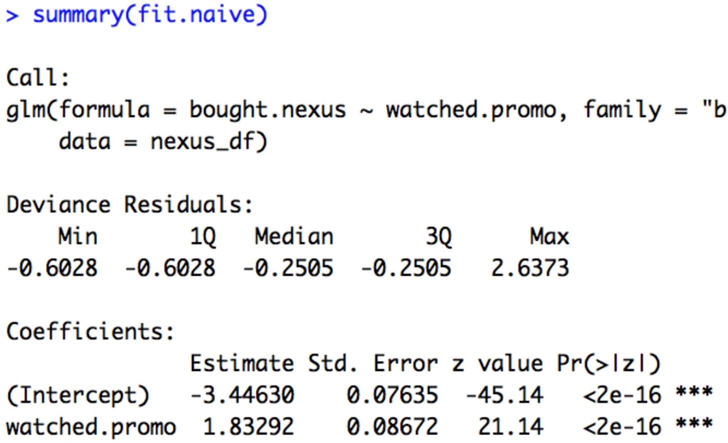
I. Propensity Scores

Intuitively, we would like to control for those factors which affect the probability of watching the video. Implicitly, we would be concerned that precisely the same factors which make people watch more videos would also make them more likely to buy the product.

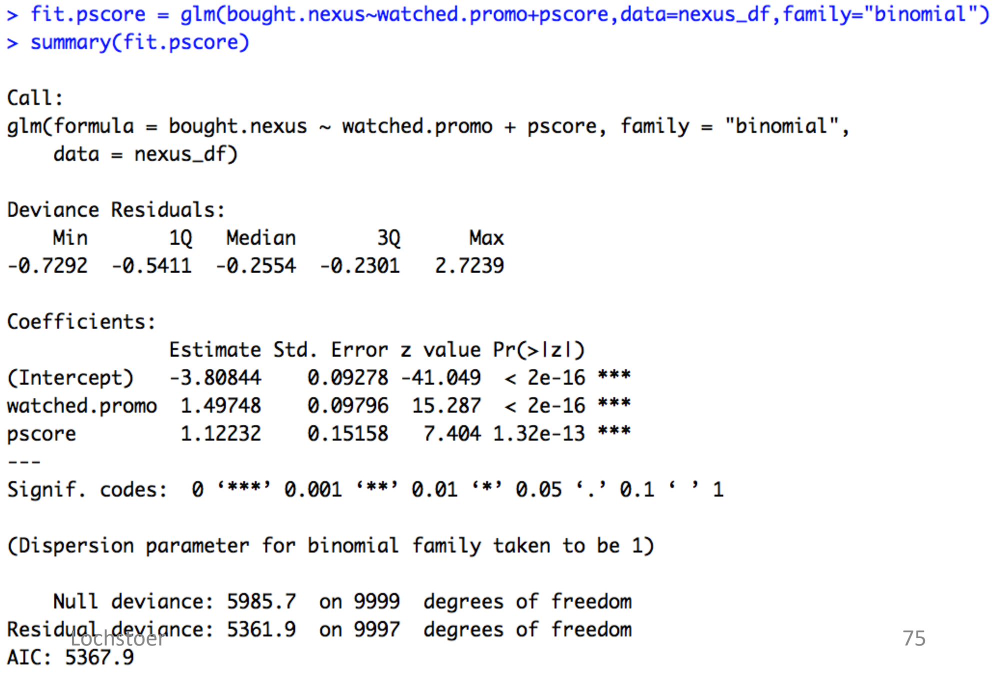
This would suggest that we would overestimate the causal effect by simply comparing those who watched with those who didn't watch. In other words, even if those people who watched the video had not seen it, we might expect them to be more likely, on average, to purchase the product.

How can we control for this? Just put the propensity score in the model of buy/no buy!

I. Propensity Scores



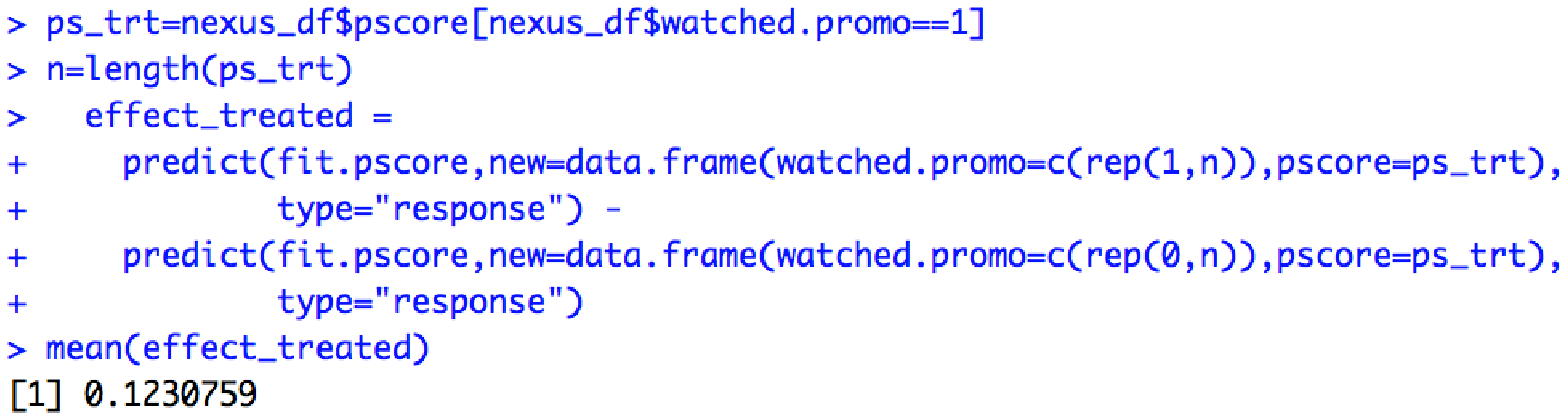
```
> summary(fit.naive)

Call:
glm(formula = bought.nexus ~ watched.promo, family = "binomial",
    data = nexus_df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6028  -0.6028  -0.2505  -0.2505   2.6373

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -3.44630    0.07635  -45.14   <2e-16 ***
watched.promo  1.83292    0.08672   21.14   <2e-16 ***
```



```
> fit.pscore = glm(bought.nexus~watched.promo+pscore,data=nexus_df,family="binomial")
> summary(fit.pscore)

Call:
glm(formula = bought.nexus ~ watched.promo + pscore, family = "binomial",
    data = nexus_df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.7292  -0.5411  -0.2554  -0.2301   2.7239

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -3.80844    0.09278 -41.049  < 2e-16 ***
watched.promo  1.49748    0.09796  15.287  < 2e-16 ***
pscore         1.12232    0.15158   7.404 1.32e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5985.7  on 9999  degrees of freedom
Residual deviance: 5361.9  on 9997  degrees of freedom
AIC: 5367.9
```

I. Propensity Scores

Now, let’s compute the effect for those who were “treated,” i.e. those who watched the promo video.



```
> ps_trt=nexus_df$pscore[nexus_df$watched.promo==1]
> n=length(ps_trt)
>   effect_treated =
+     predict(fit.pscore,new=data.frame(watched.promo=c(rep(1,n)),pscore=ps_trt),
+             type="response") -
+     predict(fit.pscore,new=data.frame(watched.promo=c(rep(0,n)),pscore=ps_trt),
+             type="response")
> mean(effect_treated)
[1] 0.1230759
```

I. Propensity Scores

Are there any “costs” of using a propensity score?

Yes, there is less information than if our assignment to treatments were made at random as we have to control for the propensity score in the logistic regression.

How do we know that the propensity score approach works? We are assuming that, after controlling for the variables used in forming the propensity score, there are no other systematic differences between the control and treatment groups in factors related to the treatment effect.

Clearly, we never know for sure. If we build a propensity score leaving out age and electronics videos, it certainly will produce a treatment effect estimate closer to the “naïve” estimate. (.129 instead of .123).