Introduction to Rossi's 402 Slides

"I have made this longer than usual, because I lack the time to make it short." - Pascal

I had the time!

A lot of work has gone into simplifying and stripping down to the essence. This means that the slides will require careful reading. They are designed to be self-contained but efficient! I hope you enjoy reading them.

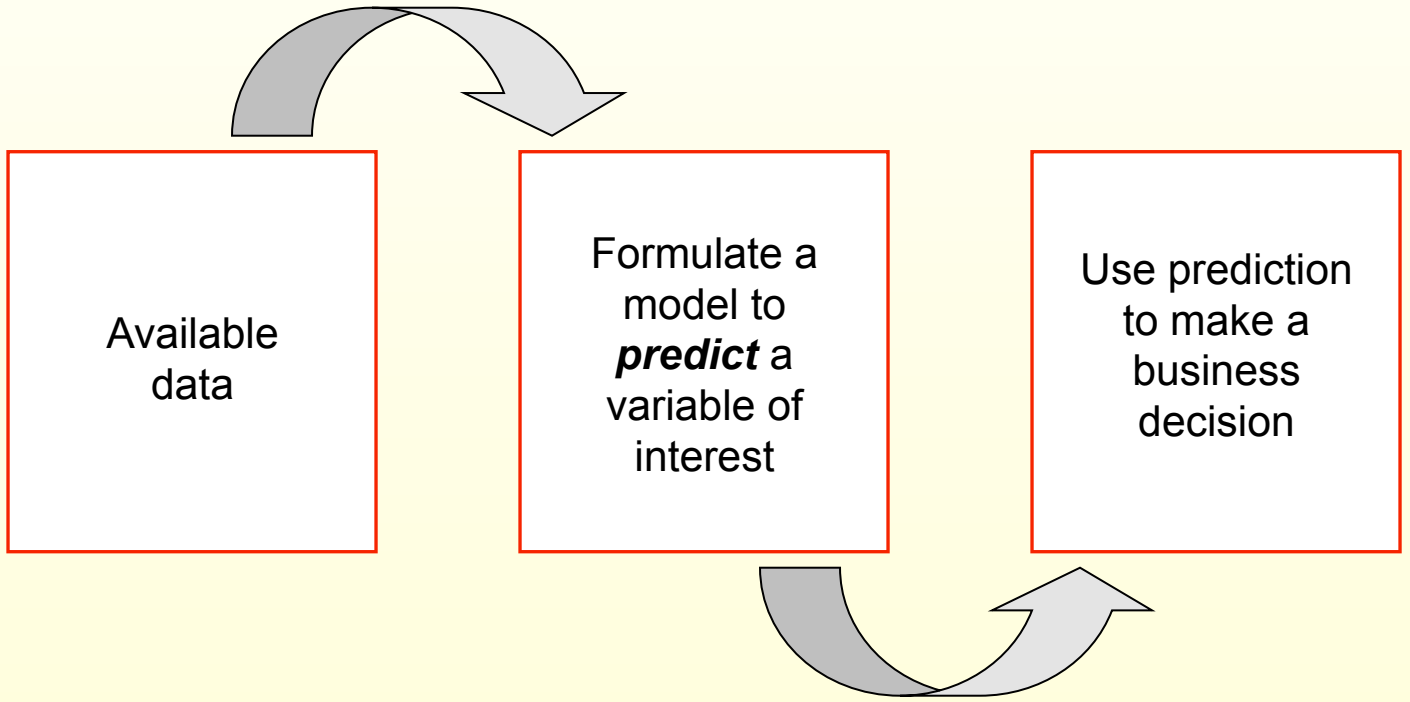
Read them three times:

1. Before class (write questions in margin)
2. During class
3. After class

I. Introduction to the Linear Regression Model

a. Conditional Prediction
b. R and R-studio
c. Mutual Fund Data
d. Linear Prediction and Least Squares
e. Relationship between b and r
f. Decomposing the variance and R^2
g. Simple Linear Regression Model
h. Sampling Distributions and Std Errors
i. Confidence Intervals
j. Hypothesis Testing
k. Prediction Intervals
l. Bias, MSE, RMSE

The basic problem:



a. Examples of Conditional Prediction

1. Optimal portfolio choice:
   - ***Predict*** future joint distribution of asset returns
   - Decision: Construct optimal portfolio (choose weights)
2. Pricing of a Product:
   - ***Predict*** sales volume response to price changes
   - Decision: what is the profit-maximizing price?

b. R, R-Studio and course package, DataAnalytics

R is a free package with versions for Windows, MAC OS, and Linux. Visit CRAN site nearest you (see above or google “CRAN”) and download and install R.

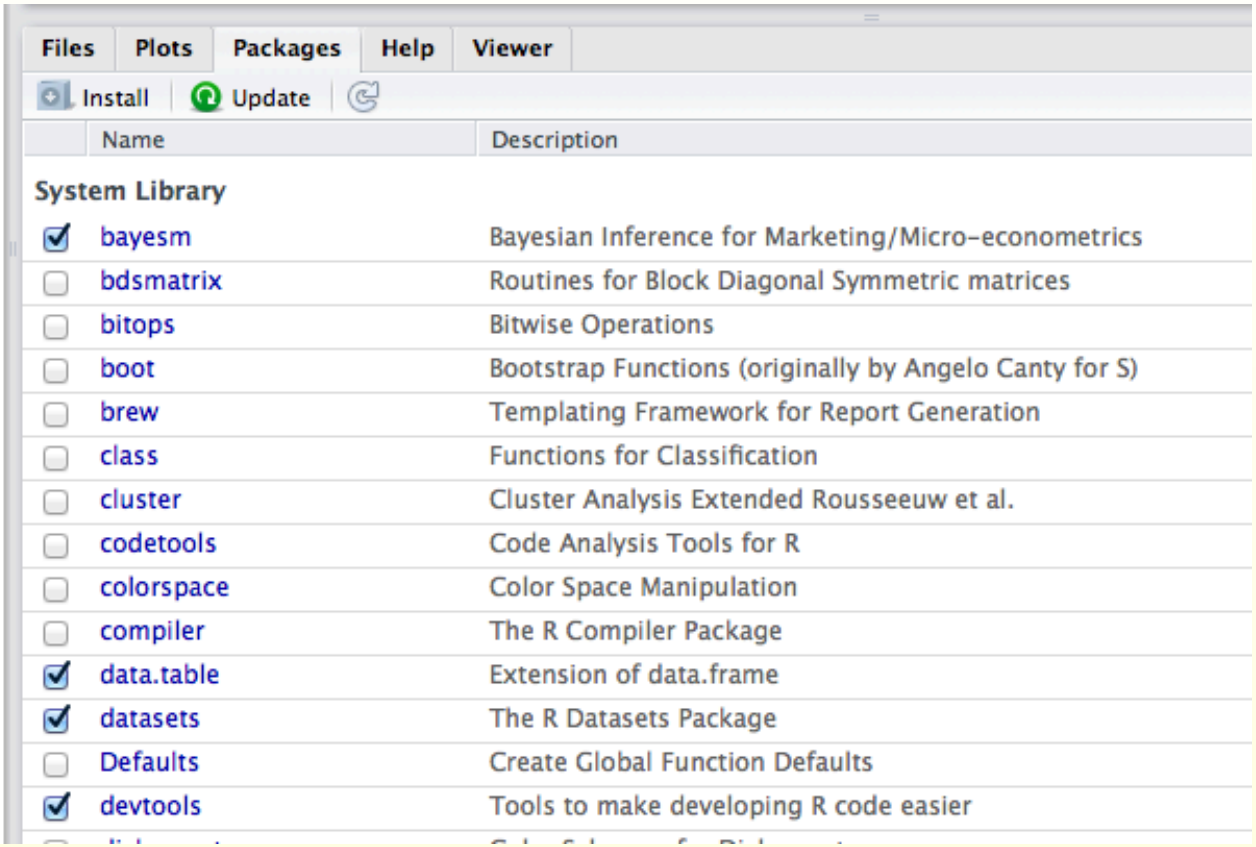
R-Studio is a programming environment for R. After installing R, download and install R-Studio.

Next you need to install the “package” made specifically for this course. This is easy to do. The package contains all of the datasets needed for the course as well as some customized functions. Steps for installation of course package:

1. Install CRAN package devtools
2. Install course package from bitbucket

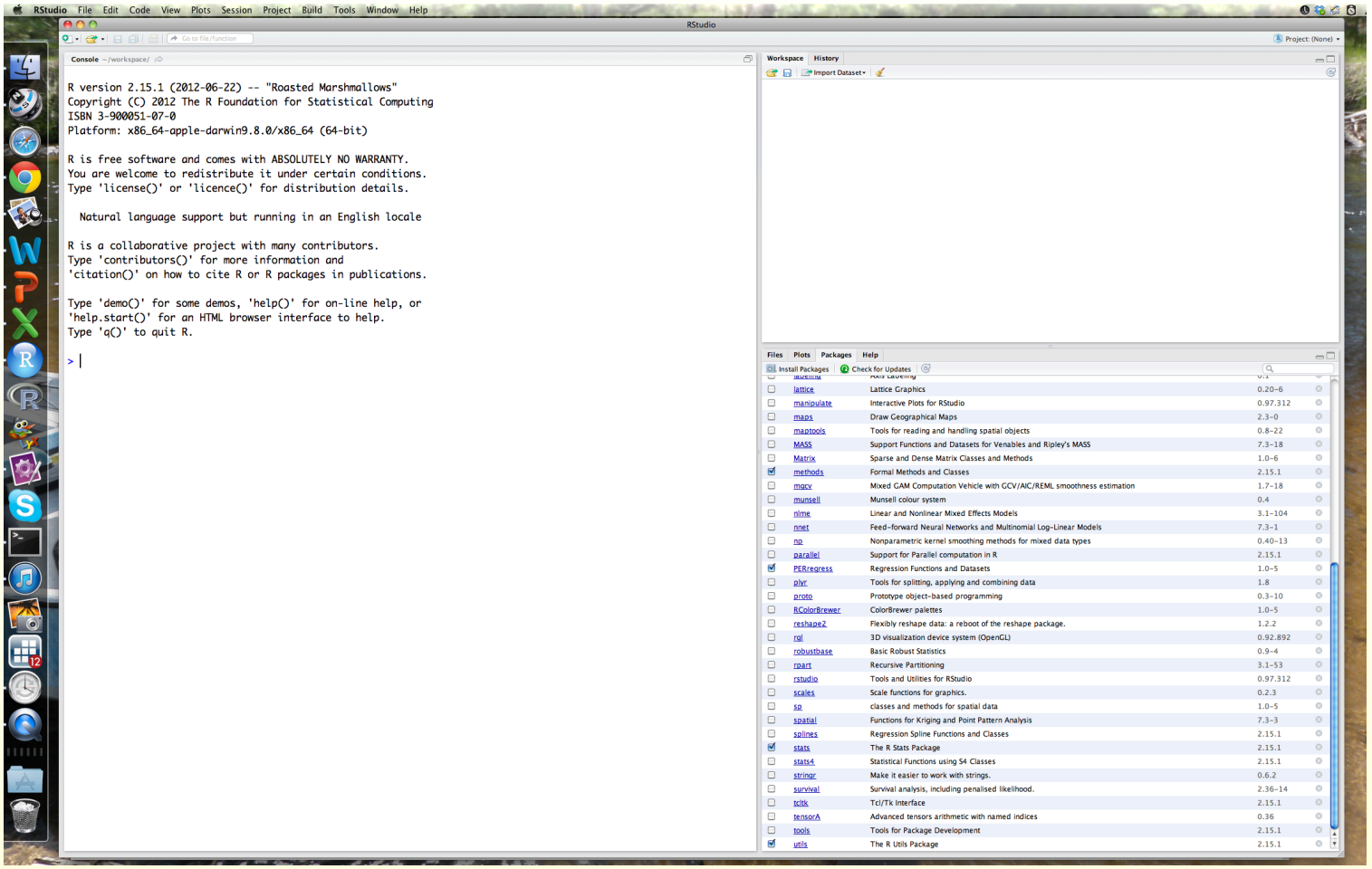
```
install_bitbucket(“perossichi/DataAnalytics”)
```

b. R, R-Studio and course package, DataAnalytics




```
> install_bitbucket("perossichi/DataAnalytics")
Downloading bitbucket repo perossichi/DataAnalytics@master
Installing DataAnalytics
```

b. The R Studio Environment



See Videos section of CCLE web site.

b. Why R?

R is a statistical programming language. It is ***the*** industry standard for all data analytic work.

R has gotten so popular because it is a very powerful and high level language which is extendible and free. R runs on all computing platforms.

There are literally 1000s of R packages which extend the functionality of R.

R's visualization and graphics capabilities are vastly better than any other statistical package or language.

b. What is the essence of R?

R is a what is called a functional programming language. Everything in R is a function.

Functions can be composed, e.g. f(g(x)) << powerful but sometimes hard to read.

R programming means to use existing R functions and make your own.

```
> data(Vanguard)
> summary(Vanguard)
      date                ticker             crsp_fundno          mtna              mret          
 Min.   :1984-06-29   Length:2294        Min.   :31184   Min.   :  -99   Min.   :-0.256365  
 1st Qu.:1995-06-30   Class :character   1st Qu.:31186   1st Qu.: 1116   1st Qu.:-0.016674  
 Median :2002-11-29   Mode  :character   Median :31217   Median : 3421   Median : 0.013213  
 Mean   :2001-09-07                      Mean   :31267   Mean   : 9389   Mean   : 0.009666  
 3rd Qu.:2008-03-31                      3rd Qu.:31277   3rd Qu.:13342   3rd Qu.: 0.037625  
 Max.   :2013-06-28                      Max.   :31462   Max.   :71789   Max.   : 0.330797  
                                                         NA's   :218                        

> mean(rnorm(1000000,mean=.01))
[1] 0.0107014
```

c. The Mutual Fund Data

The Vanguard dataset contains monthly returns on various Vanguard mutual funds.

$$R_t = \frac{\left(P_t - P_{t-1}\right) + D_t}{P_{t-1}}$$

Purchase one share at time t-1, liquidate position at time t. Earn capital gains plus any dividend payout.

Let's look at this dataset.

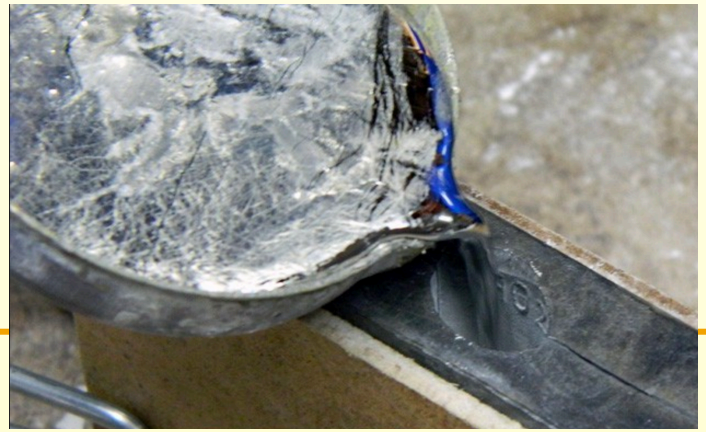
```
> data(Vanguard)
> head(Vanguard)
        date ticker crsp_fundno   mtna      mret
1 1988-04-29  VEIPX       31217     NA  0.010215
2 1988-05-31  VEIPX       31217     NA  0.027300
3 1988-06-30  VEIPX       31217 16.763  0.030535
4 1988-07-29  VEIPX       31217     NA  0.005797
5 1988-08-31  VEIPX       31217     NA -0.013449
6 1988-09-30  VEIPX       31217 28.259  0.041992
```

c. The Mutual Fund Data

There are 9 mutual funds in the dataset. The variable **`ticker`** tells us which fund the data is on. The variable **`mret`** contains the monthly returns for each fund stacked up on top of each other in one column.

Let's "unstack" or reshape our data so that there is an array or spreadsheet with one column for each fund. That is, let's create a new dataset with 10 columns (date and monthly returns for each of the funds.

To do this, we will use the R contributed package, `reshape2`, available from CRAN.



c. The Mutual Fund Data

Here is the R-code to do this:

```
library(reshape2)
data(Vanguard)
Van=Vanguard[,c(1,2,5)]   # grab relevant cols
V_reshaped=dcast(Van,date~ticker,value.var="mret")
```

```
> dim(Vanguard)
[1] 2294    5
> dim(V_reshaped)
[1] 349  10
> head(V_reshaped)
        date VEIPX VFIAX     VGENX     VGHCX VMGRX VQNPX VSMAX VTSAX VWNFX
1 1984-06-29    NA    NA -0.067921  0.002070    NA    NA    NA    NA    NA
2 1984-07-31    NA    NA -0.105381 -0.021694    NA    NA    NA    NA    NA
3 1984-08-31    NA    NA  0.186717  0.090813    NA    NA    NA    NA    NA
4 1984-09-28    NA    NA  0.016895 -0.023233    NA    NA    NA    NA    NA
5 1984-10-31    NA    NA -0.041537  0.027750    NA    NA    NA    NA    NA
6 1984-11-30    NA    NA  0.010834  0.002893    NA    NA    NA    NA    NA
```

c. The Mutual Fund Data

Now let's compute some simple summary statistics.

```
> descStat(V_reshaped)
       Mean Median    SD   IQR SE Mean 95% CI-L 95% CI-U NMissing
VEIPX 0.009  0.012 0.037 0.043   0.002    0.005    0.013       46
VFIAX 0.004  0.011 0.045 0.050   0.004   -0.003    0.011      198
VGENX 0.012  0.012 0.060 0.073   0.003    0.005    0.018        0
VGHCX 0.014  0.016 0.041 0.046   0.002    0.010    0.018        0
VMGRX 0.010  0.013 0.072 0.073   0.005    0.000    0.021      163
VQNPX 0.009  0.014 0.045 0.055   0.003    0.004    0.014       31
VSMAX 0.009  0.017 0.059 0.072   0.005    0.000    0.018      198
VTSAX 0.005  0.012 0.046 0.053   0.004   -0.003    0.012      198
VWNFX 0.009  0.014 0.043 0.048   0.002    0.005    0.014       13
Number of Observations =  349
```

Why are there missing observations?

c. The Mutual Fund Data

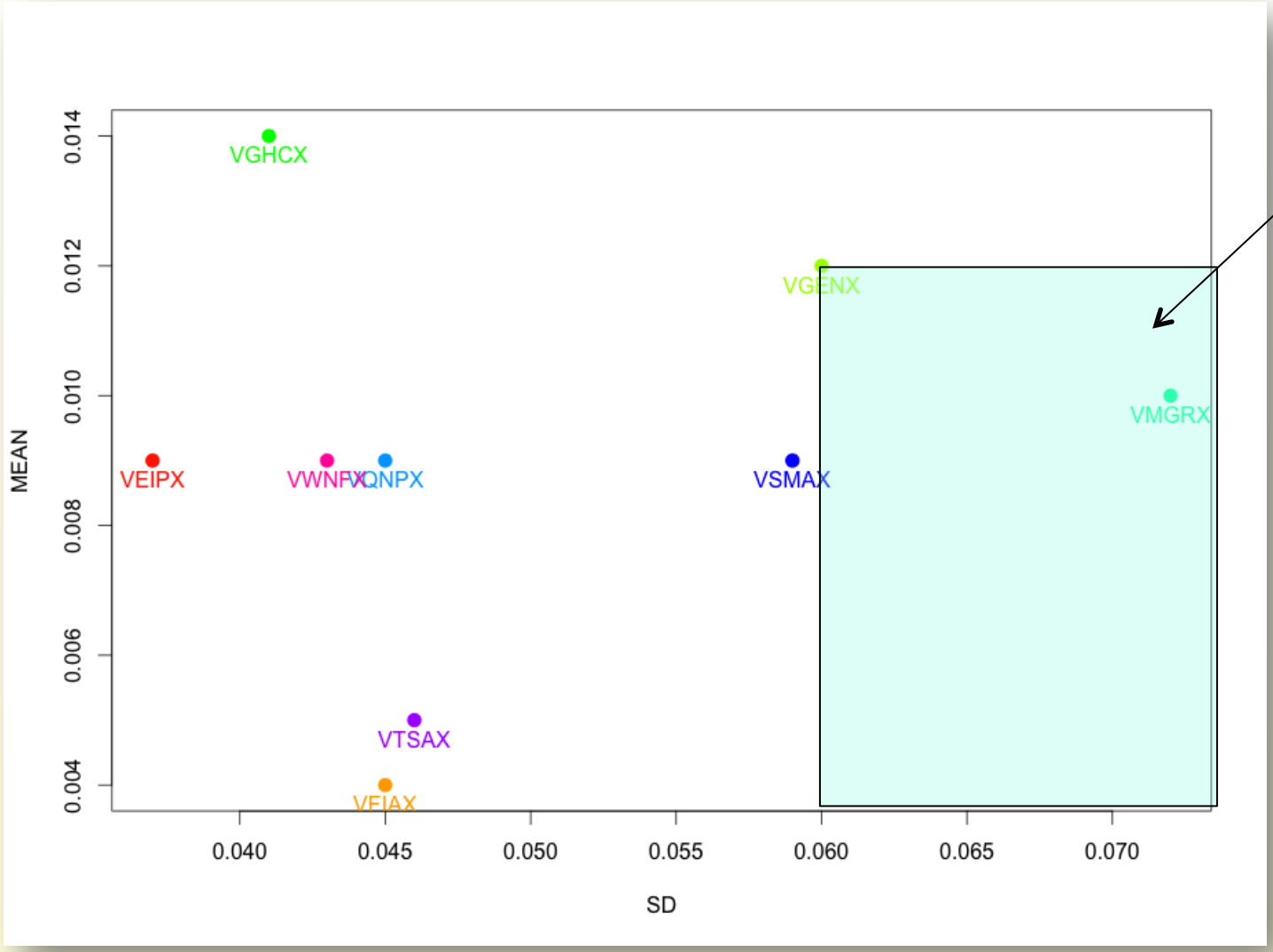
Let's plot standard deviation vs. mean.

```
mat=descStat(V_reshaped) # here we save desc stats!
SD=mat[,3]               # fetch relevant cols
MEAN=mat[,1]
funds=rownames(mat)      # retrieve fund ticker symbols

plot(SD,MEAN,pch=20,col=rainbow(length(funds)),cex=2)
         # plot SD vs. Mean with colored points
text(SD,MEAN,funds,col=rainbow(length(funds)),pos=1)
         # put the text name of the fund by each point
```

What can we say on the basis of this plot?

c. The Mutual Fund Data



VGENX dominates this region

c. The Mutual Fund Data

What about sampling error in our estimates of the mean?

	Mean	Median	SD	IQR	SE Mean	95% CI-L	95% CI-U	NMissing
VEIPX	0.009	0.012	0.037	0.043	0.002	0.005	0.013	46
VFIAX	0.004	0.011	0.045	0.050	0.004	-0.003	0.011	198
VGENX	0.012	0.012	0.060	0.073	0.003	0.005	0.018	0
VGHCX	0.014	0.016	0.041	0.046	0.002	0.010	0.018	0

$$\mathrm{Var}\left(\bar{Y}\right) = \frac{\sigma_y^2}{N} \qquad \mathrm{StdErr}\left(\bar{Y}\right) = \frac{s_y}{\sqrt{N}}$$

$$\mathrm{StdDev}\left(\bar{Y}\right) = \frac{\sigma_y}{\sqrt{N}} \qquad s_y = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left(Y_i - \bar{Y}\right)^2}$$

Why do we prefer standard deviations? Sometimes called volatilities?

c. The Mutual Fund Data

Why do we prefer standard deviations (sometimes called volatilities) to variance as a measure of spread or dispersion?

What is unusual about this data? SD > mean! SE mean is huge!

Is VEIPX better than VFIAX?

Mean much greater!
SD about the same

But, look at CI's. They overlap!

```
> descStat(V_reshaped)
       Mean Median    SD   IQR SE Mean 95% CI-L 95% CI-U NMissing
VEIPX 0.009  0.012 0.037 0.043   0.002    0.005    0.013        46
VFIAX 0.004  0.011 0.045 0.050   0.004   -0.003    0.011       198
```

d. Linear Prediction and Least Squares

Let's study the relationship between the Health Index return (VGHCX) and the overall market by doing a scatter plot of the VGHCX return on market return.

Why?

Imagine we own the index fund (market) and are thinking about moving our portfolio in the direction of the VGHCX fund. Can we improve our performance (return vs. variance) by doing so?

Let's fit a line to the cluster of points, using regression methods. We will regress VGHCX on vwretd.

VGHCX is the dependent variable which is being "explained" or regressed on the independent or predictor or explanatory variable, vwretd.

d. Linear Prediction and Least Squares

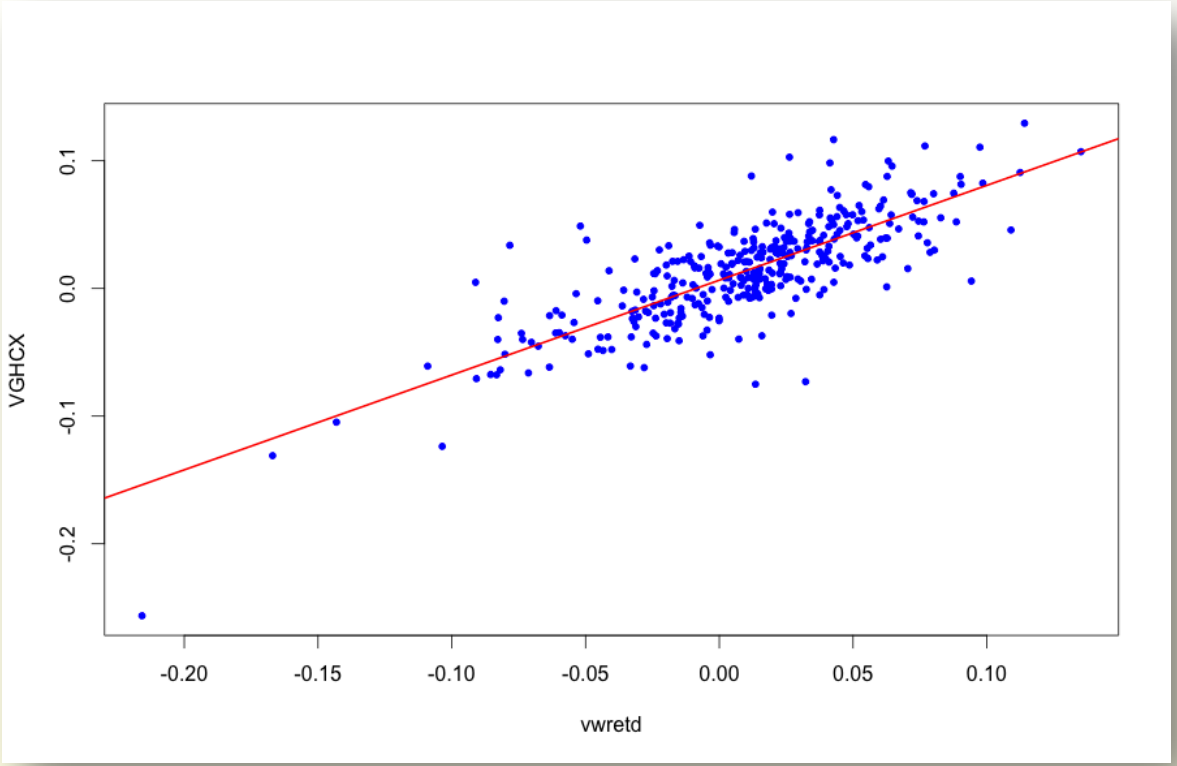
First, we have to merge the market return information into the Vanguard reshaped dataset.

A merge is also called a “join” in the database literature. For each value of the `date` variable in the Vanguard data, we want to pull the associated values from the marketRf data file for the various market indices and risk-free rates.

Let’s merge and plot the data

```
data(marketRf)
Van_mkt=merge(V_reshaped,marketRf,by="date")
with(Van_mkt,
     plot(vwretd,VGHCX,pch=20,col="blue")
)
```

d. Linear Prediction and Least Squares



Fit a line to the plot.

Dependent variable

Independent variable

```
out=lm(VGHCX~vwretd,data=Van_mkt)
abline(out$coef,col="red",lwd=2)
```

d. Linear Prediction and Least Squares

```
> summary(out)

Call:
lm(formula = VGHCX ~ vwretd, data = Van_mkt)

Residuals:
      Min        1Q    Median        3Q       Max
-0.103496 -0.014496 -0.000082  0.014088  0.085483

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.006528   0.001374   4.751 2.97e-06 ***
vwretd      0.743009   0.030135  24.656  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.02505 on 347 degrees of freedom
Multiple R-squared:  0.6366, Adjusted R-squared:  0.6356
F-statistic: 607.9 on 1 and 347 DF,  p-value: < 2.2e-16
```

Natural questions to ask at this point:

- How does R select a best fitting line?
- Can we say anything about the accuracy of the predictions?
- What does all that other R output mean?

d. Least Squares

A strategy for estimating the slope and intercept parameters

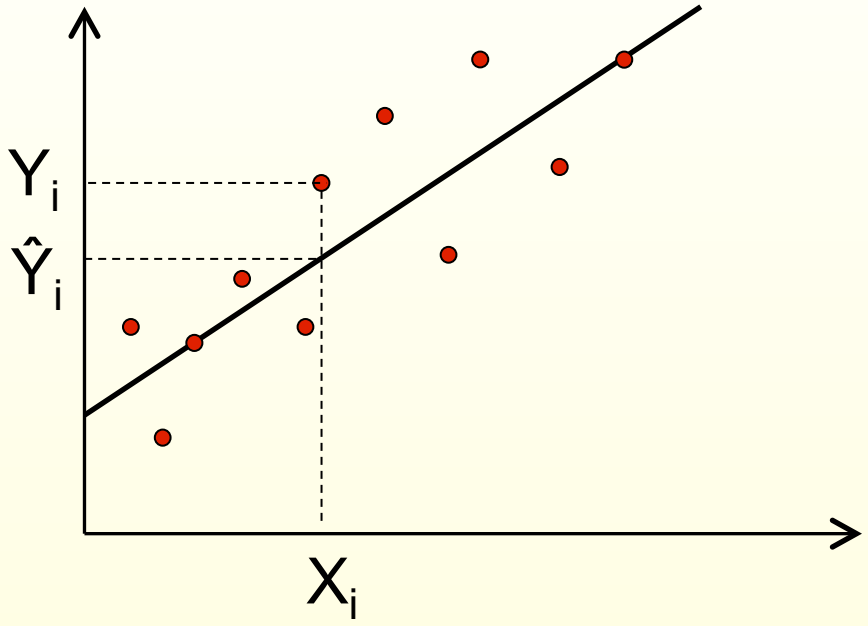
Data: We observe the data recorded as the pairs (X_i, Y_i) $i = 1,\ldots,N$

Problem: Choose a fitted line. (b_0, b_1)

A reasonable way to fit a line is to minimize the amount by which the ***fitted value*** differs from the actual value. This is called the ***residual***.

d. Least Squares: Fitted Values & Residuals

What does "**fitted value**" mean?

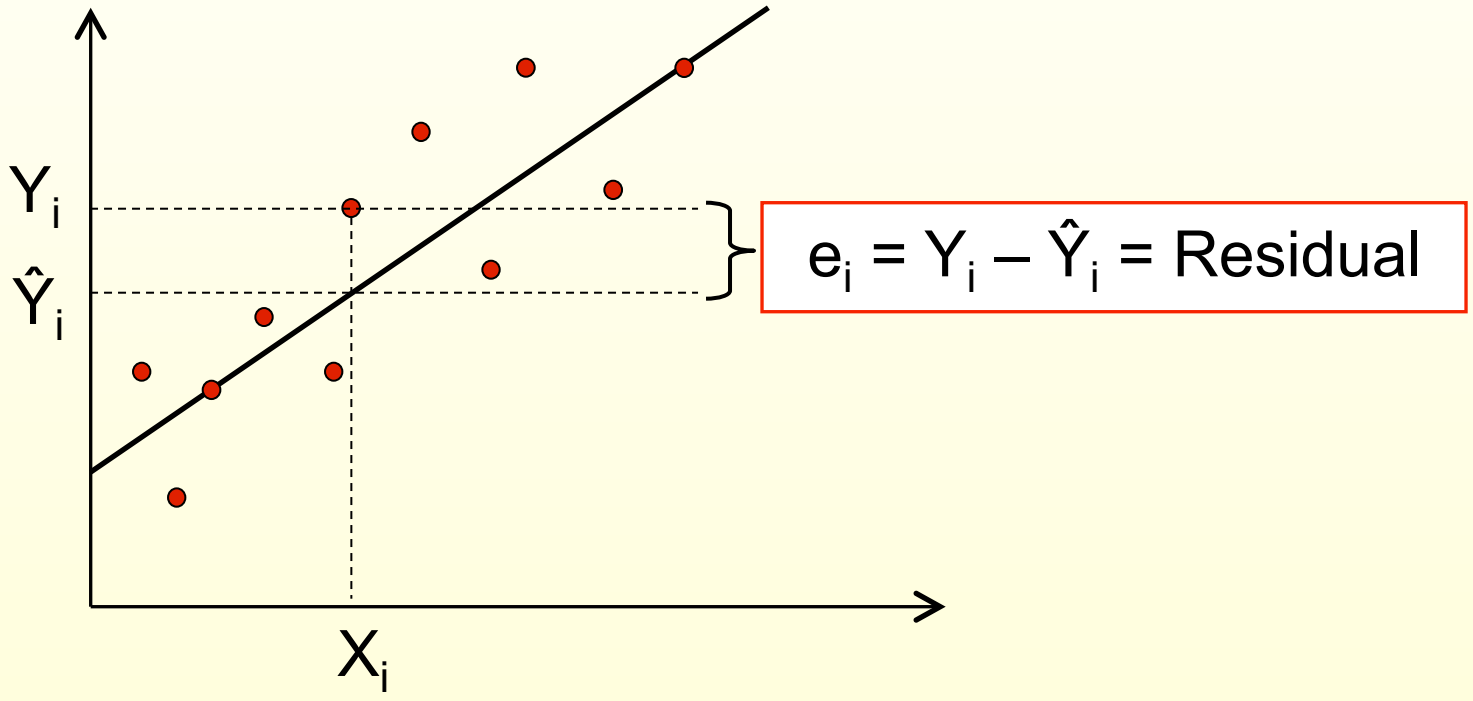


The dots are the observed values and the line represents our fitted values given by:

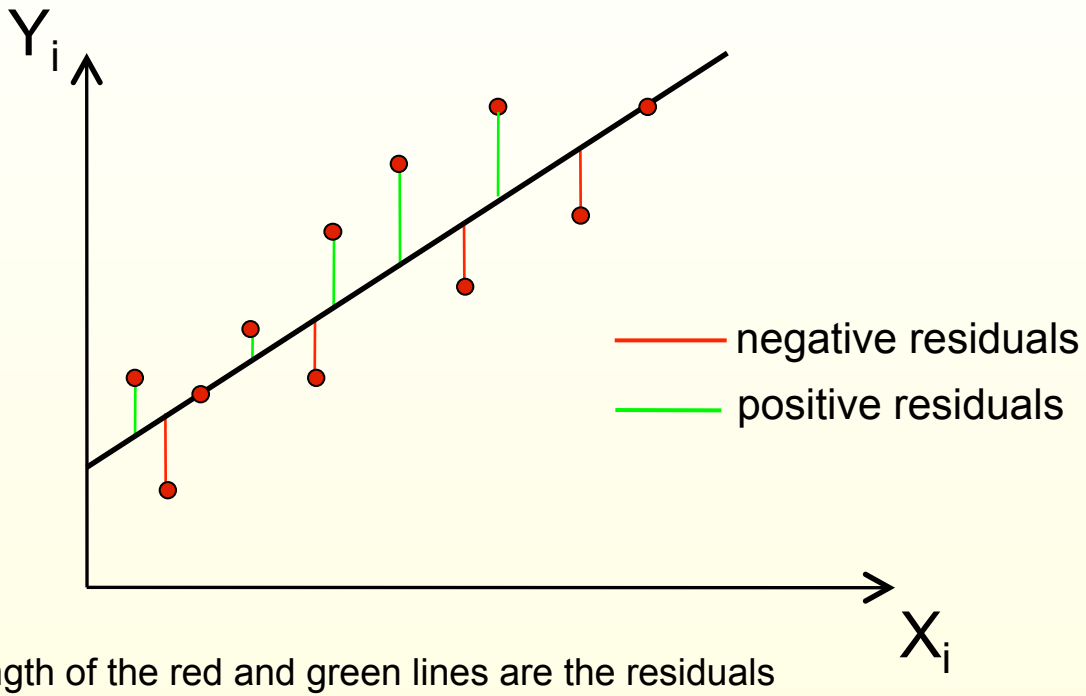
$$\hat{Y}_i = b_0 + b_1 X_i$$

d. Least Squares: Fitted Values & Residuals

What about the **residual** for the i^{th} observation?



d. Least Squares: Fitted Values & Residuals



Note: The length of the red and green lines are the residuals

The fitted value and the residual decompose the Y_i observation into two parts:

$$Y_i = \hat{Y}_i + (Y_i - \hat{Y}_i) = \hat{Y}_i + e_i$$

d. The Least Squares Criterion

Ideally, we want to minimize the size of all residuals.

We must therefore trade off between moving closer to some points and at the same time moving away from other points

A Line-Fitting "Scorecard":

i. Compute residuals
ii. Square residuals and add up
iii. Pick the best fitting line (intercept and slope)

Least Squares:

choose b_0 and b_1 to minimize

$$\sum_{i=1}^{N} e_i^2$$

d. The Least Squares Criterion

What are the formulas which do the job?

Least Squares Solution:

$$b_1 = \frac{\sum_{i=1}^{N}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{N}(X_i - \bar{X})^2} = \frac{s_{XY}}{s_X^2}$$

Where: s_{XY} = sample covariance (X, Y)

s_X^2 = sample variance of (X)

$$b_0 = \bar{Y} - b_1 \bar{X}$$

d. Intuition Behind the Least Squares Formula

The intercept:

The formula for b_0 insures that **the fitted regression line passes through the point of means** $(\overline{X}, \overline{Y})$.

If you put in the average value of X, the least squares prediction is the average value of Y.

If we substitute in for the intercept using the LS formula, we obtain:

$$\hat{Y} - \overline{Y} = b_1 (X - \overline{X})$$

If X is above the mean, then b_1 tells us how much to scale this deviation above the mean to produce a forecast of Y relative to the mean of Y

d. Intuition Behind the Least Squares Formula

We can think of least squares as a two-part process:

i. Plot the point of means

ii. Find a line rotating through that point which has the smallest sum of squared residuals

There are many possible lines that pass thru the point of means.

The least squares approach suggests that one line is the best.
Can we understand this formula by application of a fundamental intuition derived from prediction?

Quick Review: Correlation

A intuition for the slope formula can be developed but it is more complicated. Let's review the basic concept of a correlation first.

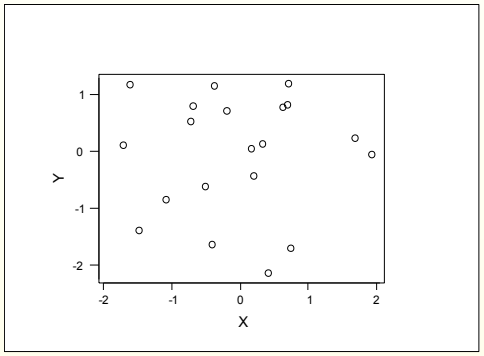
The sample coefficient between two variables is defined as:

$$r_{xy} = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum (X_i - \bar{X})^2 \sum (Y_i - \bar{Y})^2}} = \frac{s_{XY}}{s_X s_Y}$$

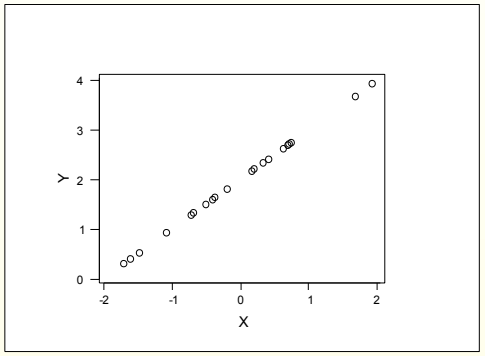
Remember: the correlation coefficient is a unitless measure of linear association between two variables.

Quick Review: Correlation

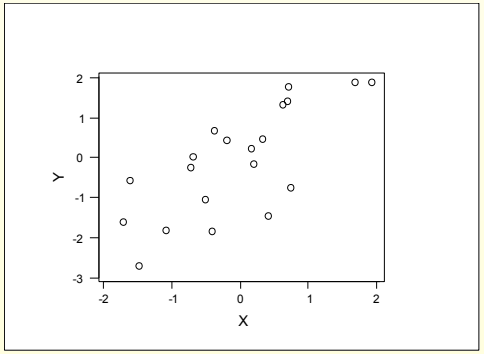
Examples of various samples with varying degrees of correlation



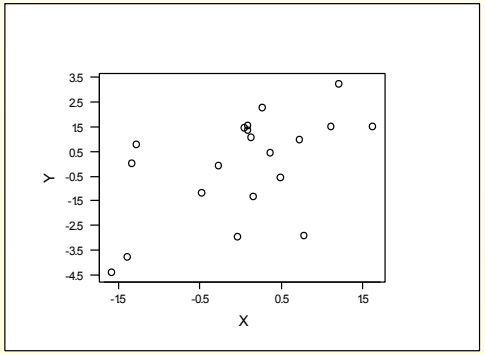
r=0



r=1



r=.75



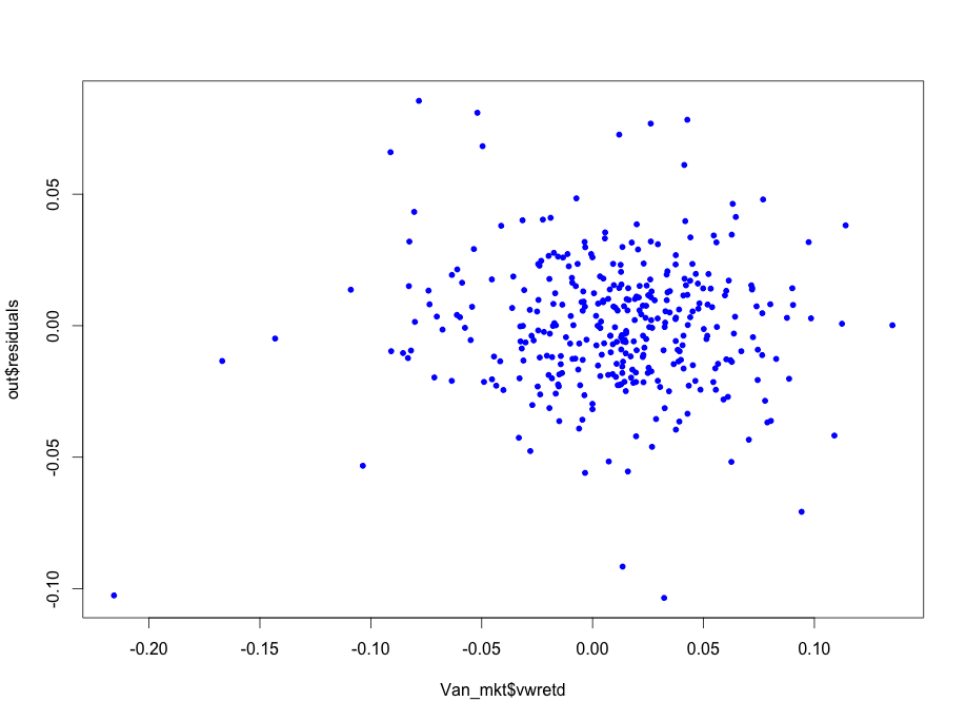
r=.5

d. Intuition Behind the Least Squares Formula

What is the intuition for the LS formula for the slope, b_1?

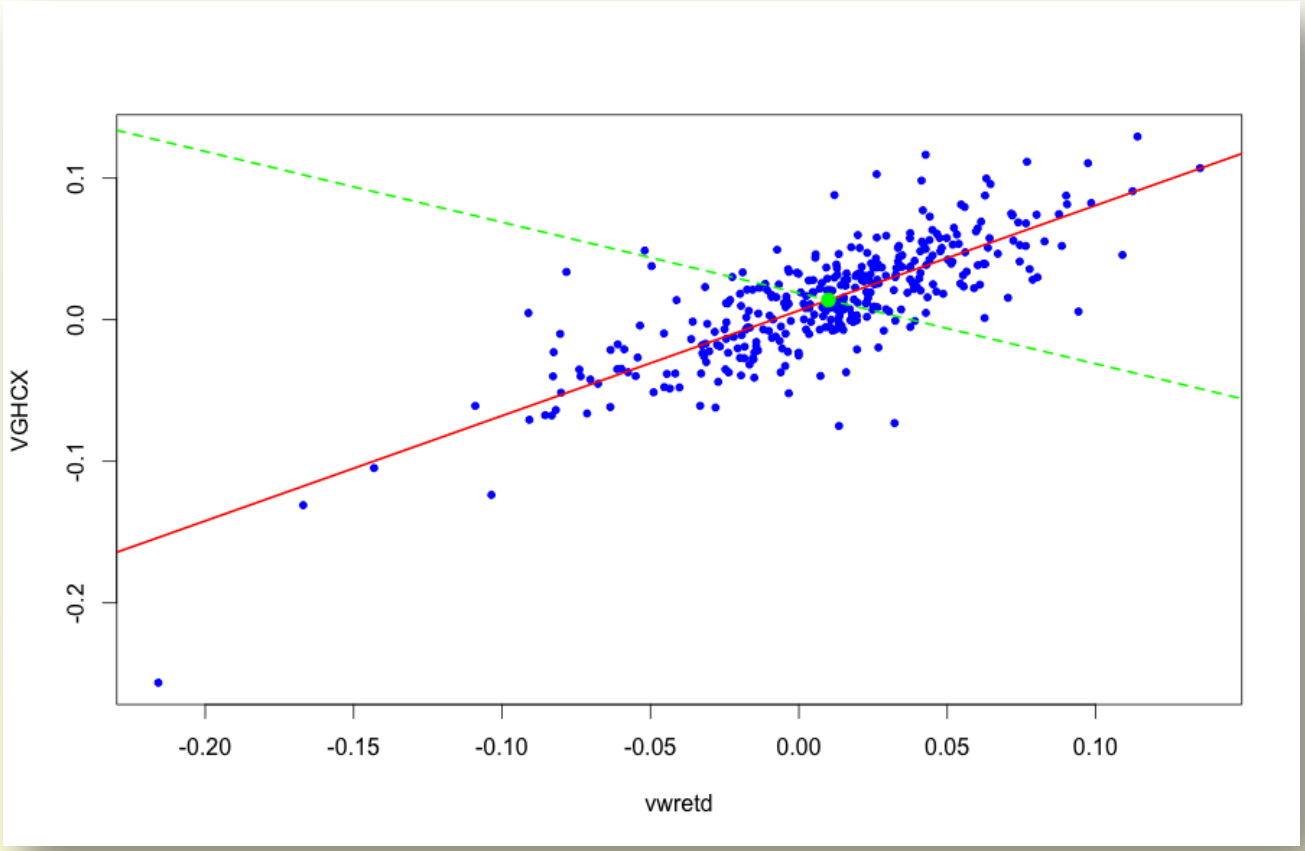
Residuals are uncorrelated with X.

```
> mean(out$residuals)
[1] 1.957059e-19
> cor(out$residuals,Van_mkt$vwretd)
[1] 3.35836e-17
```



d. Intuition Behind the Least Squares Formula

Consider a poor-fitting line – for this line there will be a correlation between X and e.



d. Intuition Behind the Least Squares Formula

As long as the correlation between e and x is non-zero, we could always adjust our prediction rule to do better

i.e. need to exploit all of the predictive power available in the X values and put this into $\hat{Y}$, **leaving no "Xness" in the residuals.**

In summary: the following decomposition of each observation is made using the fitted and residual values:

$$Y = \hat{Y} + e$$

"made from X"
corr($\hat{Y}$,X) = 1

unrelated to X
Corr(e,X) = 0

d. Intuition Behind the Least Squares Formula

The Fundamental Properties of Optimal Prediction:

- The optimal prediction of Y for an "average" or representative X should be the "average" or representative value of Y
- Prediction errors should be unrelated to the information used to formulate the predictions

For linear models:

- $\text{If } X = \bar{X},\ \hat{Y} = \bar{Y}$
- $\text{corr}\left(Y - \hat{Y}, X\right) = \text{corr}\left(e, X\right) = 0$

e. The Relationship Between b and r

b_1, can be written as the ratio of the sample covariance to the sample variance of X:

$$b_1 = \frac{s_{XY}}{s_X^2}$$

Close to, but not the same as the sample correlation.

Recall that b must "convert" the units of x into the units of y and is expressed as units of y per unit of x. However the sample correlation coefficient is unit-less. The relationship is given by:

$$b_1 = \frac{s_{xy}}{s_x^2} = r_{xy} \times \frac{s_y}{s_x}$$

f. Decomposing the Variance – The ANOVA Table

We now know that:

- $\sum e_i = 0 \quad (\text{implies } \bar{e} = 0, \text{ as well!})$
- corr(e,X) =0
- corr($\hat{Y}$,e) = 0

We can now use these factors to decompose the total variance of Y:

$$\text{Var}(Y) = \text{Var}(\hat{Y}) + \text{Var}(e) + 2\,\text{cov}(\hat{Y},e)$$

or,

$$\text{Var}(Y) = \text{Var}(\hat{Y}) + \text{Var}(e)$$

What is true for the average square (variance) is also true for the the sum of total squares,

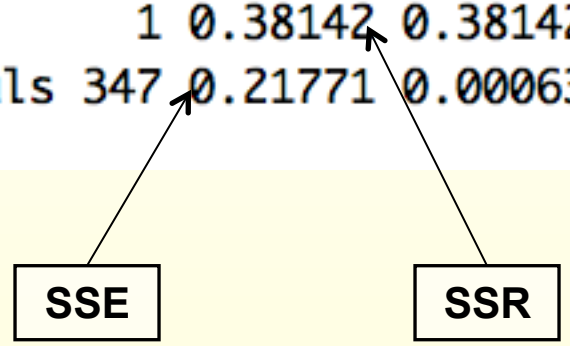
$$\underbrace{\sum_{i=1}^{N}(Y_i - \bar{Y})^2}_{\text{Total Sum of Squares SST}} = \underbrace{\sum_{i=1}^{N}(\hat{Y}_i - \bar{Y})^2}_{\text{Regression SS SSR}} + \underbrace{\sum_{i=1}^{N} e_i^2}_{\text{Error SS SSE}}$$

f. Decomposing the Variance – The ANOVA Table

Let's find this on the R printout

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

Response: VGHCX
           Df  Sum Sq Mean Sq F value    Pr(>F)
vwretd      1 0.38142 0.38142  607.92 < 2.2e-16 ***
Residuals 347 0.21771 0.00063
---
```



f. A Goodness of Fit Measure: R^2

We have a good fit if:

- SSR is large
- SSE is small
- Fit would be perfect if SST = SSR

To summarize how close SSR is to SST, we define the **coefficient of determination:**

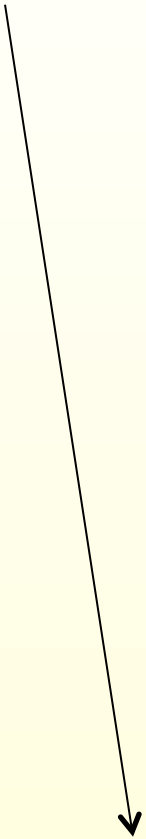
$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

Two interpretations:

i. Percentage of Variation in Y explained by X
ii. Square of correlation (hence “r” squared) – not obvious

f. A Goodness of Fit Measure: R^2

R^2 on the R printout



```
> summary(out)

Call:
lm(formula = VGHCX ~ vwretd, data = Van_mkt)

Residuals:
      Min        1Q    Median        3Q       Max
-0.103496 -0.014496 -0.000082  0.014088  0.085483

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.006528   0.001374   4.751 2.97e-06 ***
vwretd      0.743009   0.030135  24.656  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02505 on 347 degrees of freedom
Multiple R-squared:  0.6366, Adjusted R-squared:  0.6356
F-statistic: 607.9 on 1 and 347 DF,  p-value: < 2.2e-16
```

f. Misuse of R^2

R-squared is often mis-used.

Some establish arbitrary benchmarks for “high” values. For example, most regard values of over .8 as “high.”

“high” values of R^2 are often associated with claims that the model is:

1. adequate or correctly specified – “valid”
2. highly accurate for prediction

Unfortunately, neither is correct. More later!

g. The Simple Linear Regression Model

The power of statistical inference comes from the ability to make precise statements about the accuracy of the forecasts and estimates. In order to do this, we must postulate a **probability model**.

The **Simple Linear Regression Model:**

$$Y = \underbrace{\beta_0 + \beta_1 X}_{\text{Part of Y related to X}} + \underbrace{\varepsilon}_{\text{Part of Y independent of X}}$$

assumptions regarding error term?

g. The Simple Linear Regression Model

For convenience, we assume

$$E\left[\varepsilon\right] = 0$$

The size of ε is measured by its standard deviation

$$\text{StdDev}\left(\varepsilon\right) = \sigma_{\varepsilon}$$

The "systematic" part of the regression is given by $\beta_0 + \beta_1 X$

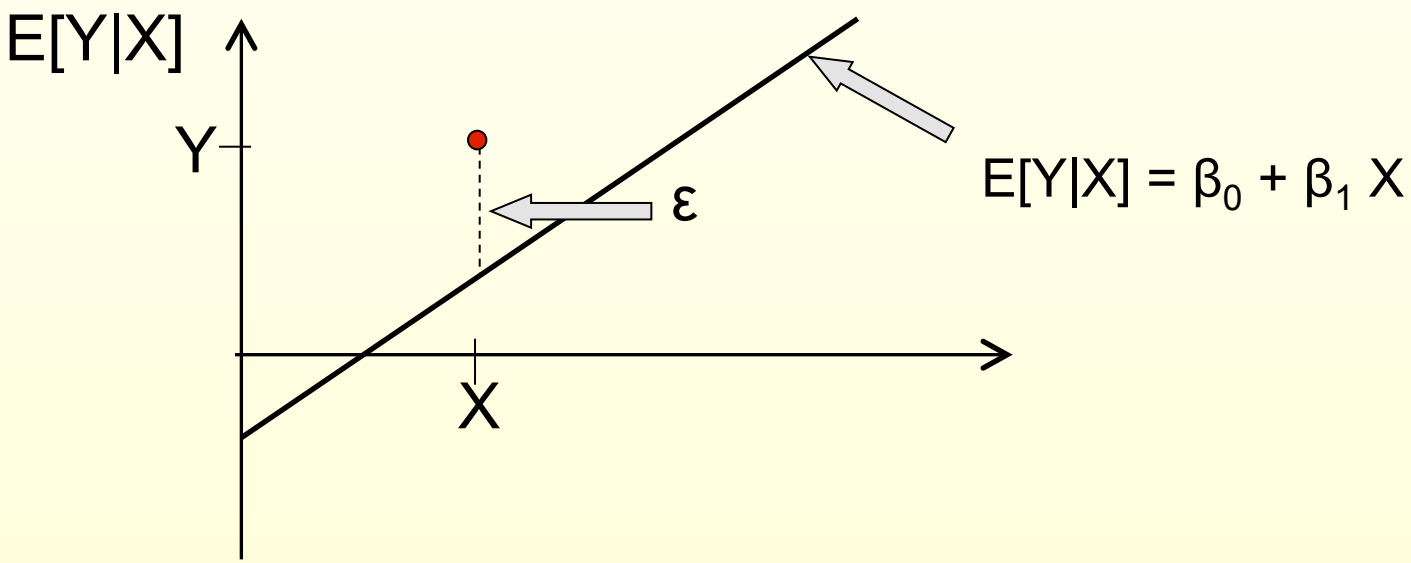
We can interpret this as the "true" regression line:

$$E\left[Y \mid X\right] = \beta_0 + \beta_1 X$$

read "*the conditional expectation of Y given X*"

g. The Simple Linear Regression Model

Think of E[Y|X] as the average return on the VGHCX when the market return is X. In some months, VGHCX return is larger than this conditional mean and in some months it is lower. The error term represents the influence of factors other than market (sometimes call idiosyncratic factors)



g. The Simple Linear Regression Model

What distribution should we use for ε ?

Justifications for using the normal distribution:

- It's the only distribution I know of!
- It works!
- The sum of many RVs often has a normal looking distribution (Central Limit Theorem)

Now the model becomes:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

$$\varepsilon \sim N\left(0, \sigma_\varepsilon^2\right)$$

Mean of ε is 0

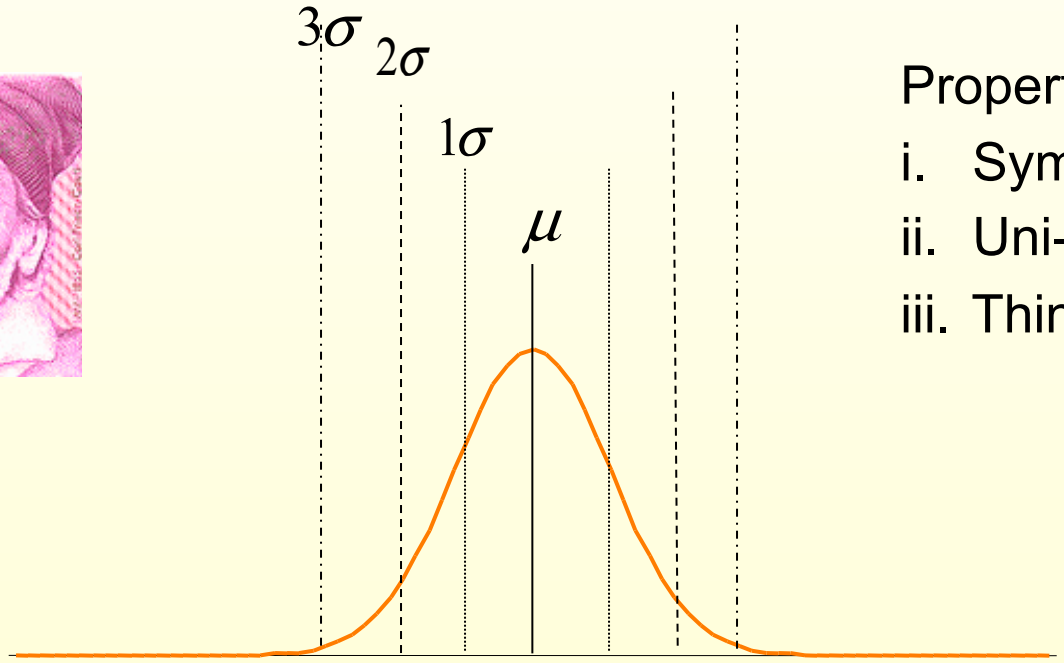
Sometimes Y is above the line; sometimes below it

Remember we think of σ as the average size of the error term

Quick Review of Normal Distribution

Remember the relationship between σ and where the normal distribution puts its mass.

$Pr(\mu - \mathbf{1}\sigma < X < \mu + \mathbf{1}\sigma)$	.68	**One Sigma**
$Pr(\mu - \mathbf{2}\sigma < X < \mu + \mathbf{2}\sigma)$	.9544	**Two Sigma**
$Pr(\mu - \mathbf{3}\sigma < X < \mu + \mathbf{3}\sigma)$	.997	**Three Sigma**



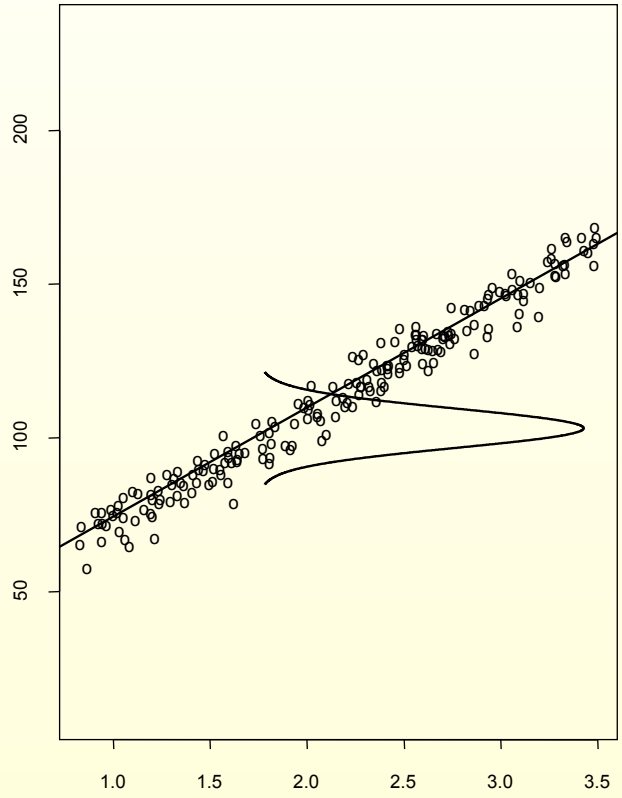
Properties of Normal:

i. Symmetric
ii. Uni-modal
iii. Thin-tails

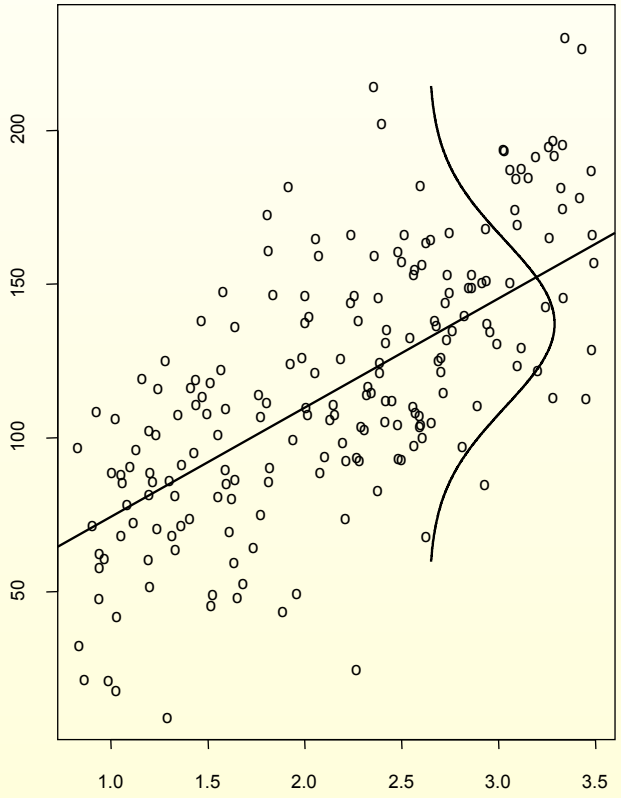
g. The Simple Linear Regression Model

Let's look at the role of σ in determining the dispersion of points about the true regression line.

σ small / ε small



σ large / ε large



g. Summary of Simple Linear Regression

Assume that all observations are drawn from the simple linear regression model and that errors on those observations are independent.

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

$$\varepsilon_i \sim \text{iid } N(0, \sigma_\varepsilon^2)$$

MODEL

Independent and Identically distributed

The SLR has 3 basic parameters.

$$\beta_0,\ \beta_1,\ \sigma_\varepsilon$$

Describe linear pattern

Size of errors or variation about the line

g. Key Characteristics of Linear Regression Model

Three Key Characteristics:

1. Mean of Y is linear in X

2. Error terms (deviations from line) are normally distributed (few deviations > 3 sd away from line)

3. Error terms have constant variance.

g. Estimation of σ^2

Recall that, $\varepsilon_i \sim \text{iid}N\left(0,\sigma_\varepsilon^2\right)$

and that σ drives the width of the prediction intervals

$$\sigma_\varepsilon^2 = \text{Var}(\varepsilon_i) = \text{E}\left[(\varepsilon_i - \text{E}\left[\varepsilon_i\right])^2\right] = \text{E}\left[\varepsilon_i^2\right]$$

One sensible strategy would be to estimate this population average of squared errors with the sample average squared residuals

$$\hat{\sigma}^2 = \frac{1}{N}\sum_{i=1}^{N} e_i^2$$

g. Estimation of σ^2

However, this is not an unbiased estimator of σ^2. We have to alter the denominator slightly:

$$s^2 = \frac{1}{N-2}\sum_{i=1}^{N} e_i^2 = \frac{SSE}{N-2}$$

# of degrees of freedom are reduced by 2 because 2 have been "used up" in the estimation of b_0 and b_1

Usually we want to use s (not s^2). Why? s is in the same units as Y.

$$s = \sqrt{\frac{SSE}{N-2}} = \text{standard error of the regression}$$

Recall: "standard error" means estimated standard deviation.

g. Conditional Distributions vs. Marginal Distributions

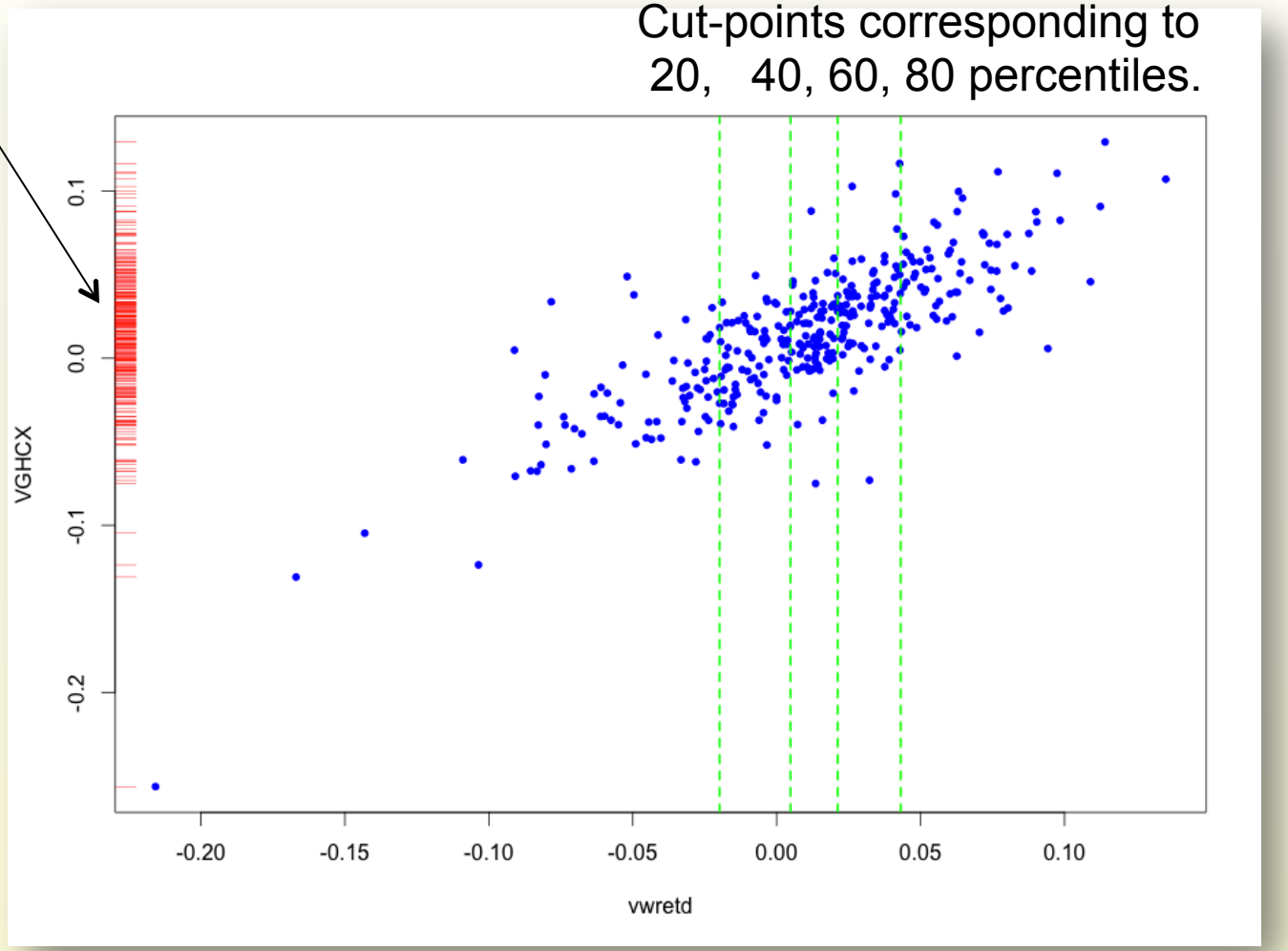
Regression models are really all about modeling the conditional distribution of Y given X.

Why are conditional distributions important? We want to develop models for forecasting. What we are doing is exploiting the information in the conditional distribution of Y given X.

The conditional distribution is obtained by "slicing" the point cloud in the scattergram to obtain the distribution of Y conditional on various ranges of X values.

g. Conditional Distributions vs. Marginal Distributions

Marginal distribution of VGHCX



g. Conditional Distributions vs. Marginal Distributions

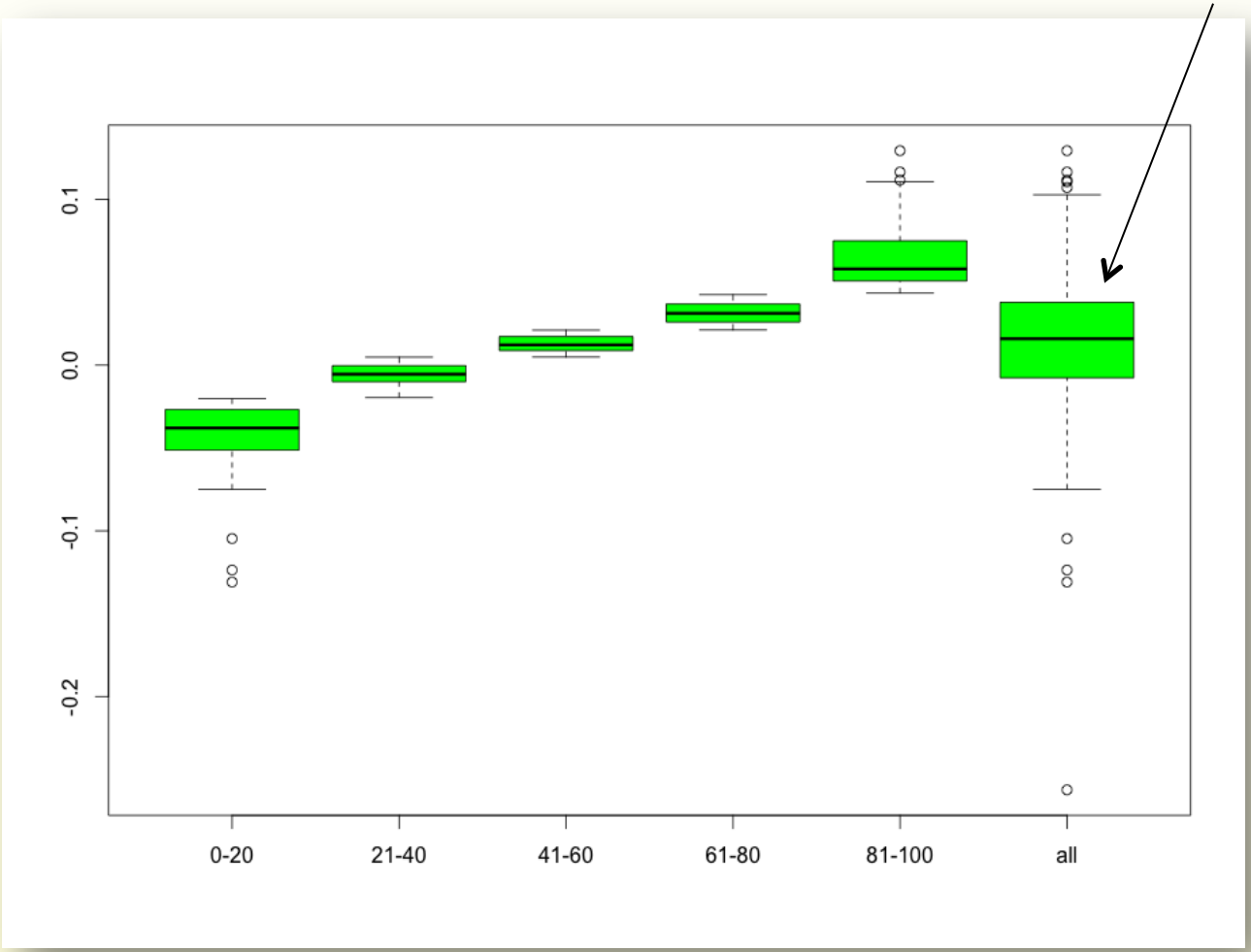
R code for plot:

```
plot(vwretd,Van_mkt$VGHCX,pch=20,col="blue")
# add rug on left side
rug(Van_mkt$VGHCX,side=2,col="red")
# paint slices
qvec = quantile(Van_mkt$vwretd,probs=c(.2,.4,.6,.8))
abline(v=qvec,lty=2,lwd=2,col="green")
```

g. Conditional Distributions vs. Marginal Distributions

Conditional distributions of VGCHX

Marginal Dist of VGCHX



R code for plot:

```
cat=cut(Van_mkt$VGHCX,
    breaks=c(min(Van_mkt$VGHCX),qvec,max(Van_mkt$VGHCX)))
cat=c(as.numeric(cat),rep(6,length(cat)))
cat=factor(cat,
    labels=c("0-20","21-40","41-60","61-80","81-100","all"))
# repeat whole data on bottom to get marginal
VGHCXbig=rep(Van_mkt$VGHCX,2)
boxplot(VGHCXbig~cat,col="green")
```

g. Conditional Distributions vs. Marginal Distributions

This suggests two general points:

- If X has no forecasting power, then
  the marginal and conditionals will be the same.

- If X has some forecasting information or power, then

  conditional means will be different than the marginal or overall mean

  and

  conditional standard deviation(s) of Y given X will be less than the marginal standard deviation of Y.

g. Conditional Distributions vs. Marginal Distributions

The conditional distribution of Y given X:

$$Y \mid X = x \sim \quad N\left(\beta_0 + \beta_1 x, \sigma^2_{Y|X}\right)$$

This equation should be read as "*the conditional distribution of Y given X is a normal distribution with mean* $\beta_0 + \beta_1 X$ *and standard deviation* $\sigma_{Y|X}$ "
Note: to assume that Y is normal conditional on X does not mean that X has to be normally distributed!

In general,

$$\sigma_Y > \sigma_{Y|X} \text{ if X and Y are related.}$$

or

Marginal Variance of Y > Conditional Variance of Y | X

if X is worth knowing!

g. Conditional Distributions vs. Marginal Distributions

To see that this is equivalent to the expression involving the error term, ε, let's compute the conditional mean and variance.

If $Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$

since E[ε] = 0,

$$E\left[Y \middle| X = x\right] = \beta_0 + \beta_1 x$$

$$Var\left(Y \middle| X = x\right) = Var\left(\varepsilon\right) = \sigma_\varepsilon^2$$

so

$$\boxed{\sigma_\varepsilon = \sigma_{Y|X}}$$

g. Conditional Distributions vs. Marginal Distributions

Let's compute the marginal variance of Y and contrast this to the conditional variance of Y | X!

$$\begin{aligned}\mathrm{Var}\left(Y_i\right) &= \mathrm{Var}\left(\beta_0 + \beta_1 X_i + \varepsilon_i\right) = \mathrm{Var}\left(\beta_1 X_i + \varepsilon_i\right) \\ &= \beta_1^2 \mathrm{Var}\left(X_i\right) + \mathrm{Var}\left(\varepsilon_i\right) = \beta_1^2 \mathrm{Var}\left(X_i\right) + \sigma_\varepsilon^2\end{aligned}$$

$$\mathrm{Var}\left(Y_i\right) = \beta_1^2 \mathrm{Var}\left(X_i\right) + \sigma_\varepsilon^2 > \sigma_\varepsilon^2 = \sigma_{Y|X}^2$$

unless $\beta_1 = 0$!

g. What does this say about the VGHCX fund?

A common benchmark for performance is the market index.

Let's look at the relationship between returns on the market index and VGCX.

- Does VGHCX outperform the market?
- Does the VGHCX do better in up markets than down markets?

These are questions about the distribution of VGHCX given the Market or

$$R_{VGHCX,t} = \alpha + \beta R_{M,t} + \varepsilon_t$$

If ε_t is normal, then this is a standard regression model.

The **Market Model** assumes that $R_{M,t}$ is normal

g. What does this say about the VGHCX fund?

What are we hoping to see in the conditional distribution of VGHCX given the market?

market timing ←---→ non-linear relationship

Stock-picking ←--→ higher return than market – move in lock-step

g. What does this say about the VGHCX fund?

How do we relate the regression model to performance evaluation?

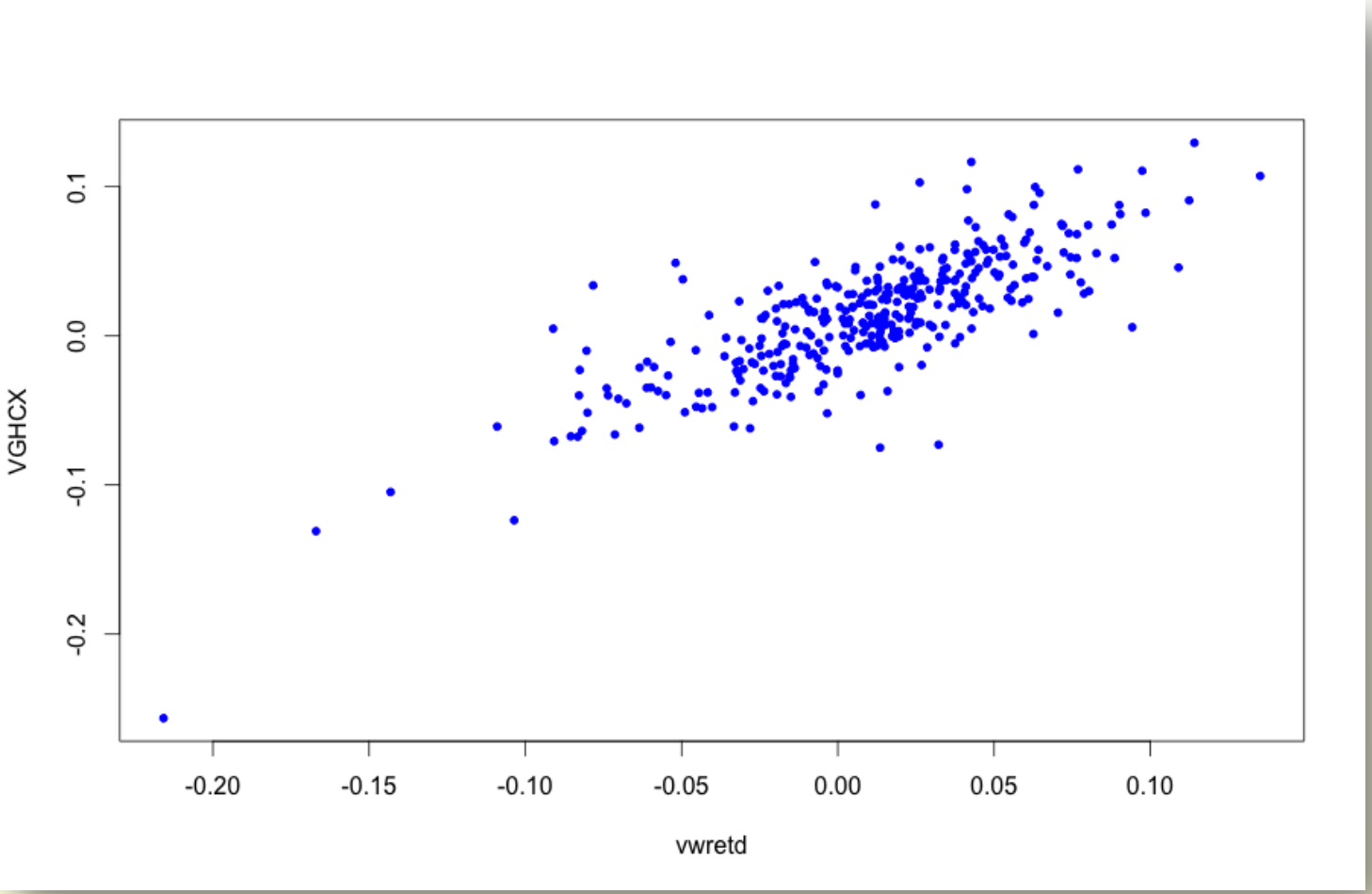
We want to measure performance relative to the market. Does the slope work as a performance measure? No!

What about the intercept? Yes! This is sometimes called Jensen's alpha.

The intercept in the above regression can be interpreted as a measure of the risk-adjusted excess return for the portfolio. We are looking for positive intercepts! What about the one above? Is it large from a substantive point of view? Is it measured precisely?

Let's review the scatterplot and regression output again.

g. What does this say about the VGHCX fund?



g. What does this say about the VGHCX fund?

```
> lmSumm(out)
Multiple Regression Analysis:
    2 regressors(including intercept) and 349 observations

lm(formula = VGHCX ~ vwretd, data = Van_mkt)

Coefficients:
            Estimate Std Error t value p value
(Intercept) 0.006528  0.001374    4.75       0
vwretd      0.743000  0.030130   24.66       0
---
Standard Error of the Regression:  0.02505
Multiple R-squared:  0.637  Adjusted R-squared:  0.636
```

What can we say about the intercept?

h. Estimation in the Simple Linear Regression Model

Recall the SLR that assumes that every observation in the dataset was generated by the model:

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \quad (i = 1, \ldots, N)$$

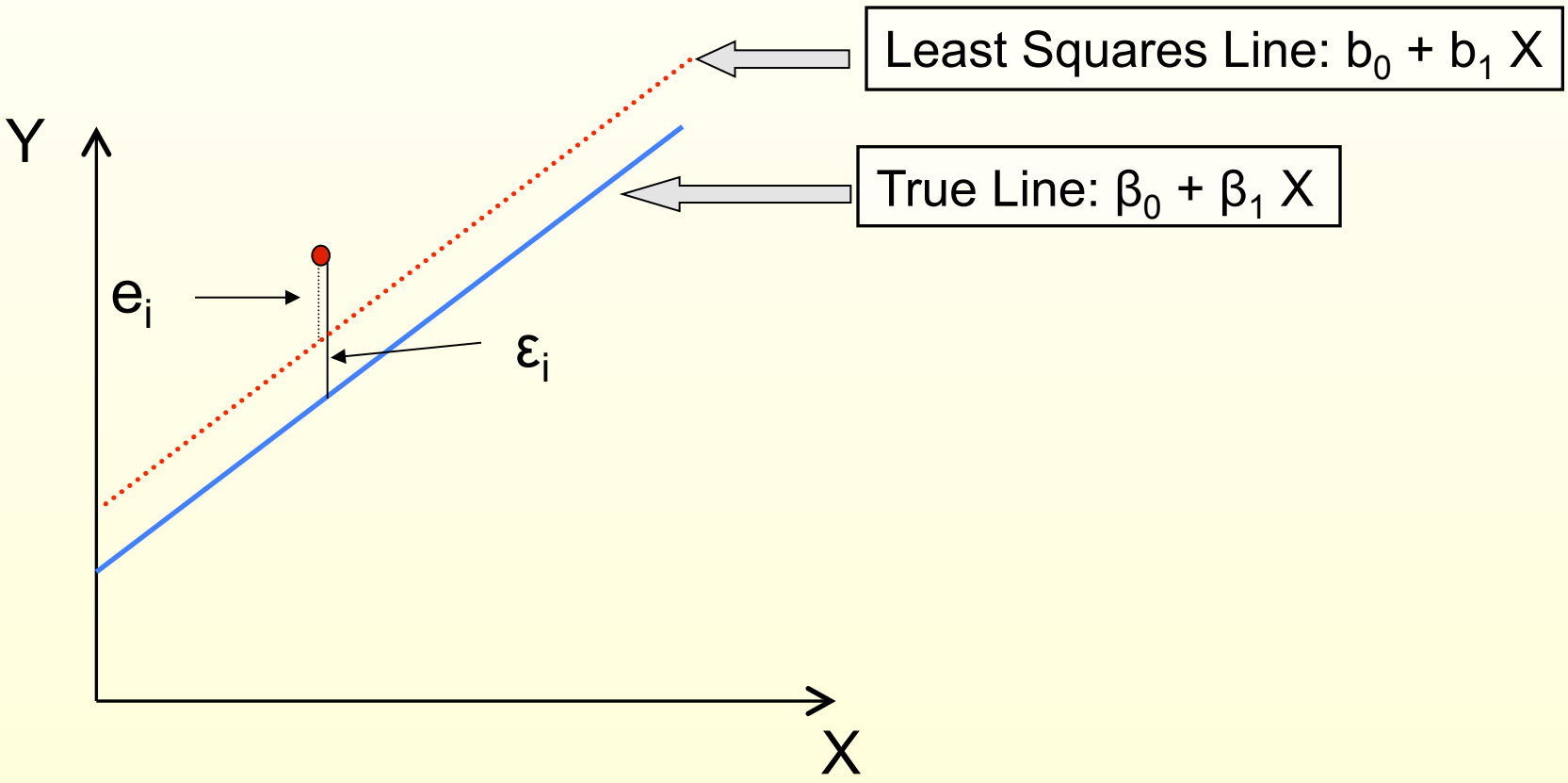
We use Least Squares to estimate β_0 and β_1 . Recall the formulas:

$$\hat{\beta}_1 = b_1 = \frac{\sum_{i=1}^{N} (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{N} (X_i - \bar{X})^2}$$

$$\hat{\beta}_0 = b_0 = \bar{Y} - b_1 \bar{X}$$

h. Estimation in the Simple Linear Regression Model

β_0 is not b_0, β_1 is not b_1 and **ε_i is not e**



h. Sampling Distribution of b_1

It is possible to derive the sampling distribution of b_1. b_1 is a weighted average of the Y values!

This distribution describes how the estimator b_1 would vary over different samples with the X values fixed.

It turns out that b_1 is normally distributed

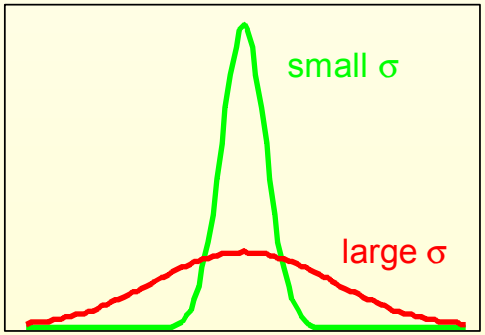
$$b_1 \sim N\left(\beta_1, \sigma^2_{b_1}\right)$$

Mean is β_1 -- unbiased

Variance of b_1

The variance term determines how close the estimate will be to the true value.

Remember: large σ_{b1} is bad!



h. Derivation of Var(b_1)

First, let's write b_1 as a linear combination of the Y's. This makes b_1 very much like a weird sort of sample average.

$$b_1 = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sum (X_i - \bar{X})^2} = \frac{\sum (X_i - \bar{X}) Y_i}{\sum (X_i - \bar{X})^2} \quad \text{(Why?)}$$

To make things easier to read, use one symbol for the denominator.

$$D = \sum (X_i - \bar{X})^2$$

Now we can see that b_1 is a linear combination of the Y's. We will call the weights, c_i.

$$b_1 = \frac{\sum (X_i - \bar{X})}{D} Y_i = \sum_{i=1}^{N} c_i Y_i \text{ where } c_i = \frac{(X_i - \bar{X})}{D}$$

h. Derivation of Var(b_1)

These are not the same weights as in the sample average (1/N vs. c_i). Let's observe some simple properties of the c_i.

Property 1: $$\sum c_i = \frac{\sum (X_i - \bar{X})}{D} = \frac{1}{D}\sum (X_i - \bar{X}) = 0$$

Weights sum up to zero. Observations farther from $\bar{X}$ receive larger weights. Remember: just because something sums to 0 doesn't mean that it is all zeroes!

Property 2: $$\sum c_i^2 = \sum \frac{(X_i - \bar{X})^2}{D^2} = \frac{1}{D^2}\sum (X_i - \bar{X})^2 = \frac{D}{D^2} = \frac{1}{D}!$$

h. Derivation of Var(b_1)

Derivation of Var(b_1)

Now that we know a few things about the c_i weights, we can easily derive the variance formula.

$$\text{Var}(b_1) = \text{Var}\left(\sum c_i Y_i \mid X\right)$$

Now we use:

i. the fact that, *given* X_i , the Y_i are independent

ii. the formula from Math-Stat prereq on the variance of a l. c. of indep r.v.s

$$\text{Var}(b_1) = \sum c_i^2 \text{Var}(Y_i \mid X_i) = \sigma^2 \sum c_i^2 = \frac{\sigma^2}{D}$$

recalling that, $D = \sum (X_i - \bar{X})^2$ we have the variance formula.

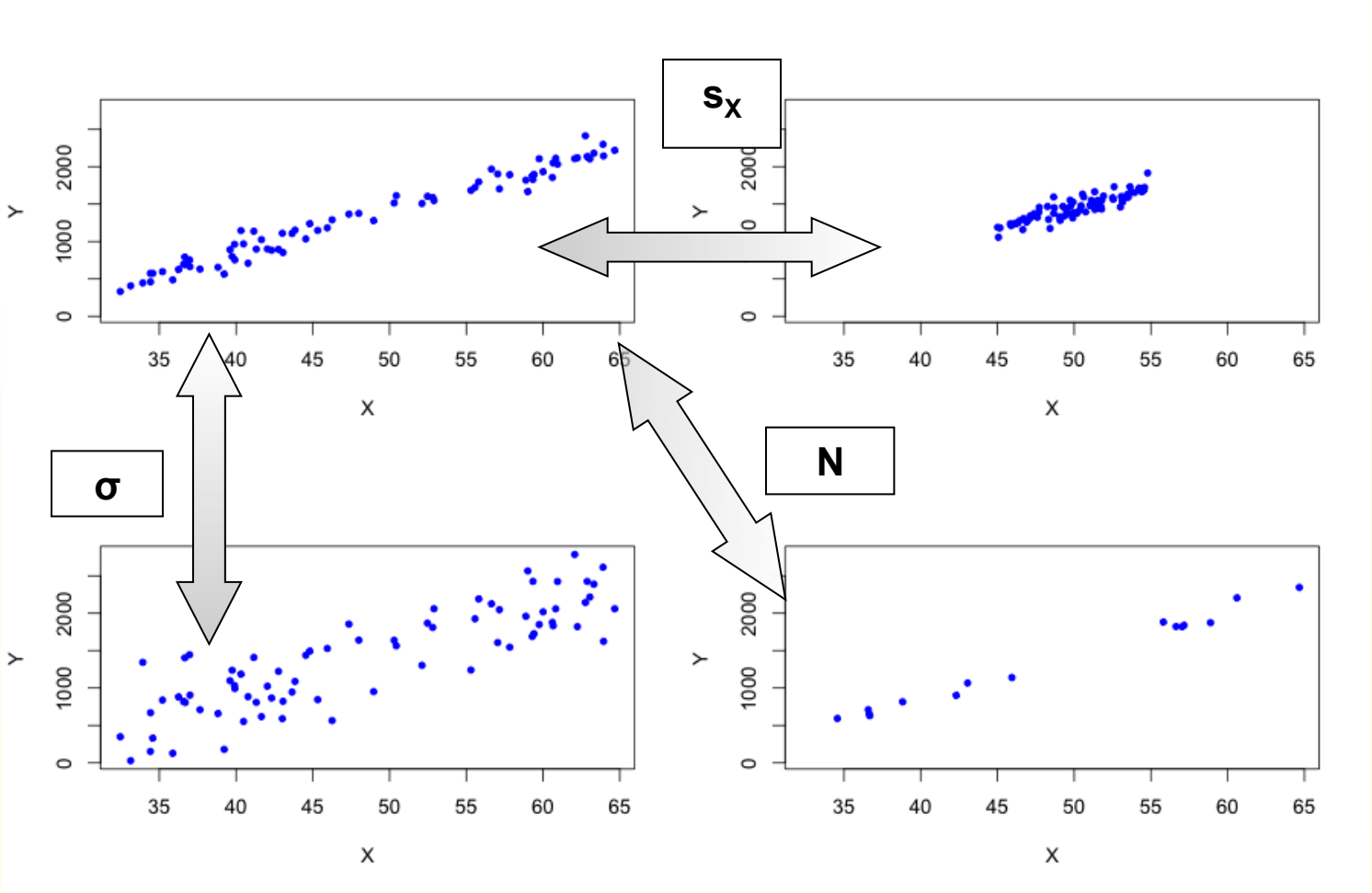
h. Sampling Distribution of b_1

What is the intuition behind the formula for $\sigma^2_{b_1}$?

- How should σ figure in the formula?
- How should N figure in the formula?
- Anything else?

$$\mathrm{Var}\left(b_1\right) = \frac{\sigma^2}{\sum_{i=1}^{N}\left(X_i - \bar{X}\right)^2} = \frac{\sigma^2}{\left(N-1\right)s_X^2}$$

h. Sampling Distribution of b_1



Note: Darker side of arrows indicates increase in parameter

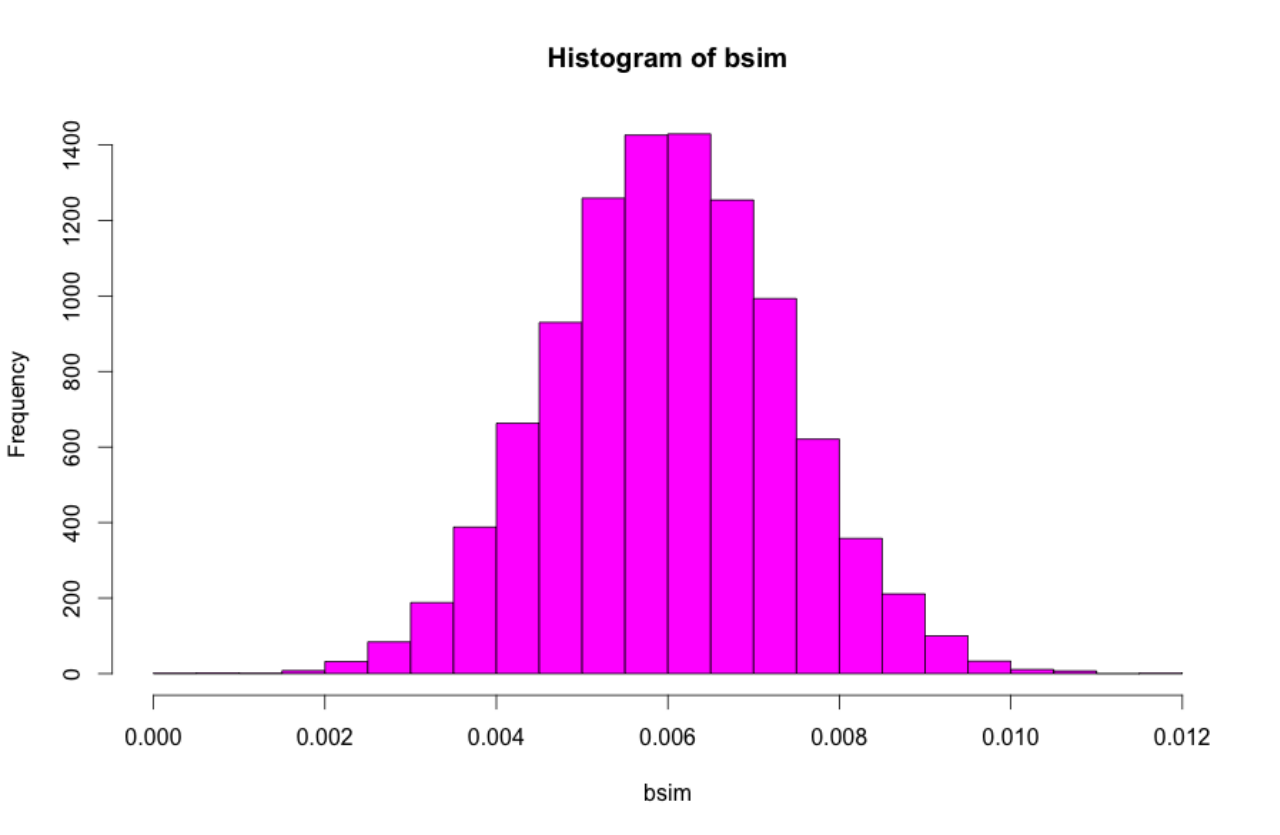
h. Using Simulation to Understand Sampling Error

Let's simulate data from a model close to what we have fit and examine the sampling distribution of b_1.

```
simreg=function(beta0,beta1,sigma,x){
  y=beta0+beta1*x+rnorm(length(x),sd=sigma)
}
x=Van_mkt$vwretd
beta0=.006; beta1=.75; sigma=0.025
nsample=10000
bsim=double(nsample)

for(i in 1:nsample){
  y=simreg(beta0,beta1,sigma,x)
  bsim[i]=lm(y~x)$coef[1]
}
hist(bsim,breaks=40,col="magenta")
```

h. Using Simulation to Understand Sampling Error




```
> mean(bsim)
[1] 0.006008205
> sd(bsim)
[1] 0.001361395
```

h. Understanding Standard Errors

If we insert our estimate of **σ**, then we have *estimated standard deviations* or **standard errors** for the least squares estimators:

$$s_{b_1} = \sqrt{\frac{s^2}{(N-1)s_X^2}}$$

S not σ

Now we can summarize the amount of information there is in the sample about the true regression line parameters.

h. Understanding Standard Errors

Where can we find the **standard errors** on the R printout?

```
> lmSumm(out)
Multiple Regression Analysis:
    2 regressors(including intercept) and 349 observations

lm(formula = VGHCX ~ vwretd, data = Van_mkt)

Coefficients:
            Estimate Std Error t value p value
(Intercept) 0.006528  0.001374    4.75       0
vwretd      0.743000  0.030130   24.66       0
---
Standard Error of the Regression:  0.02505
Multiple R-squared:  0.637  Adjusted R-squared:  0.636
```

s_{b_0}

s_{b_1}

We would like to translate the size of the standard errors into probability statements about the likely ranges of true β values.

i. Confidence Intervals

We want a margin of error in the estimation of the slope. We can use the standard errors to construct a confidence interval which provides the margin of error.

All confidence intervals are of the form: $b_1 \pm t^* s_{b_1}$

t^* is a positive number obtained from the t distribution.

So we have the estimate +/- a multiple of the standard error.

i. Confidence Intervals

To define a confidence interval, you must first set the confidence level.

We can never be completely confident that a finite interval will cover the true value. (the 100% confidence interval is everything!). So we set a confidence level. Typically, 95 per cent is used (for very large datasets a 99 per cent level should be used).

We then determine the multiple, t^*, so that there is a 95 per cent chance that the interval will cover the true value.

i. t Distribution and Confidence Intervals

Confidence intervals provide information about the range of values of the slope consistent with our data. This is much more useful than simply using the slope estimate.

An estimate without some idea of its precision is useless.

Let'd do it in R:

```
> confint(out)
                  2.5 %      97.5 %
(Intercept) 0.003825401 0.009230234
vwretd      0.683738760 0.802279083
```

i. A "Formal" Deriviation of Confidence Intervals

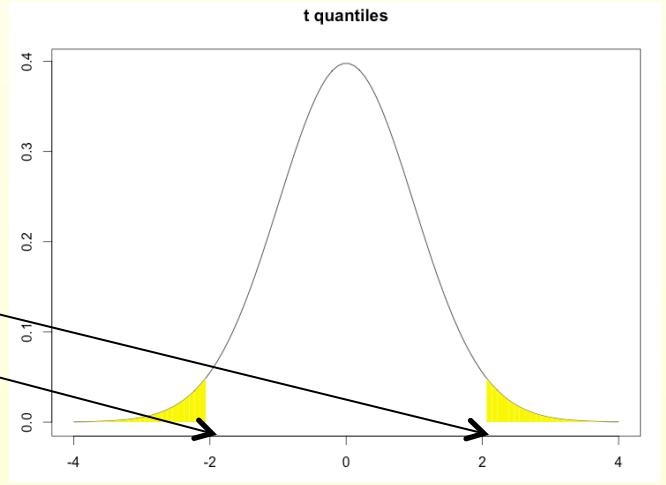
While we gave a useful informal discussion of confidence intervals, it is useful to follow a more formal definition.

t statistic has a t distribution.

$$t = \frac{b - \beta}{s_b} \sim t_\nu$$

Therefore, we can define an interval which "traps" or captures the t statistic with probability equal to the confidence level using the percentiles or quantiles of the distribution.

$$\Pr\left[t_\nu^{\alpha/2} < t < t_\nu^{1-\alpha/2} \right] = 1 - \alpha$$



i. A "Formal" Deriviation of Confidence Intervals

Or

$$\Pr\left[t_\nu^{\alpha/2} < \frac{b-\beta}{s_b} < t_\nu^{1-\alpha/2} \right] = \Pr\left[t_\nu^{\alpha/2} s_b < b-\beta < t_\nu^{1-\alpha/2} s_b \right]$$

$$= \Pr\left[-t_\nu^{\alpha/2} s_b > \beta - b > -t_\nu^{1-\alpha/2} s_b \right]$$

$$= \Pr\left[b - t_\nu^{\alpha/2} s_b > \beta > b - t_\nu^{1-\alpha/2} s_b \right]$$

$$= \Pr\left[b - t_\nu^{1-\alpha/2} s_b < \beta < b - t_\nu^{\alpha/2} s_b \right] = 1-\alpha$$

But the t-distribution is symmetric, this means $t_\nu^{1-\alpha/2} = -t_\nu^{\alpha/2}$ and we have the standard C.I. :

$$\Pr\left[b - t_\nu^{1-\alpha/2} s_b < \beta < b + t_\nu^{1-\alpha/2} s_b \right] = 1-\alpha$$

j. Hypothesis Testing

Suppose that we are interested in a specific value of the slope parameter, β_1.

This can be rephrased as a *hypothesis*

$H_0: \beta_1 = \beta_1^*$ Null (from "no effect")

vs.

$H_A: \beta_1 \neq \beta_1^*$ Alternative

For example, is there any evidence in the data to support the existence of a relationship between X and Y?

So if we want test whether X affects Y, we would test whether $\beta_1 = 0$.

j. Hypothesis Testing

How can we assess whether or not the data support or refute the null hypothesis?

We can look at our estimate of the true slope and compare it to the hypothesized value:

$$b_1 - \beta_1^* \quad \text{(discrepancy)}$$

What is wrong just using the discrepancy above? How close is close?

What do we do here? We look at the discrepancy relative to the accuracy of estimation – don't get excited unless discrepancy is large relative to accuracy.

t statistic:

$$t = \frac{b_1 - \beta_1^*}{s_{b_1}} = \frac{\text{estimate} - \text{hypo value}}{\text{std err}}$$

The basic intuition is that if the null is true then the t statistic should be small (in absolute value).

Get worried when t is large!

j. Hypothesis Testing

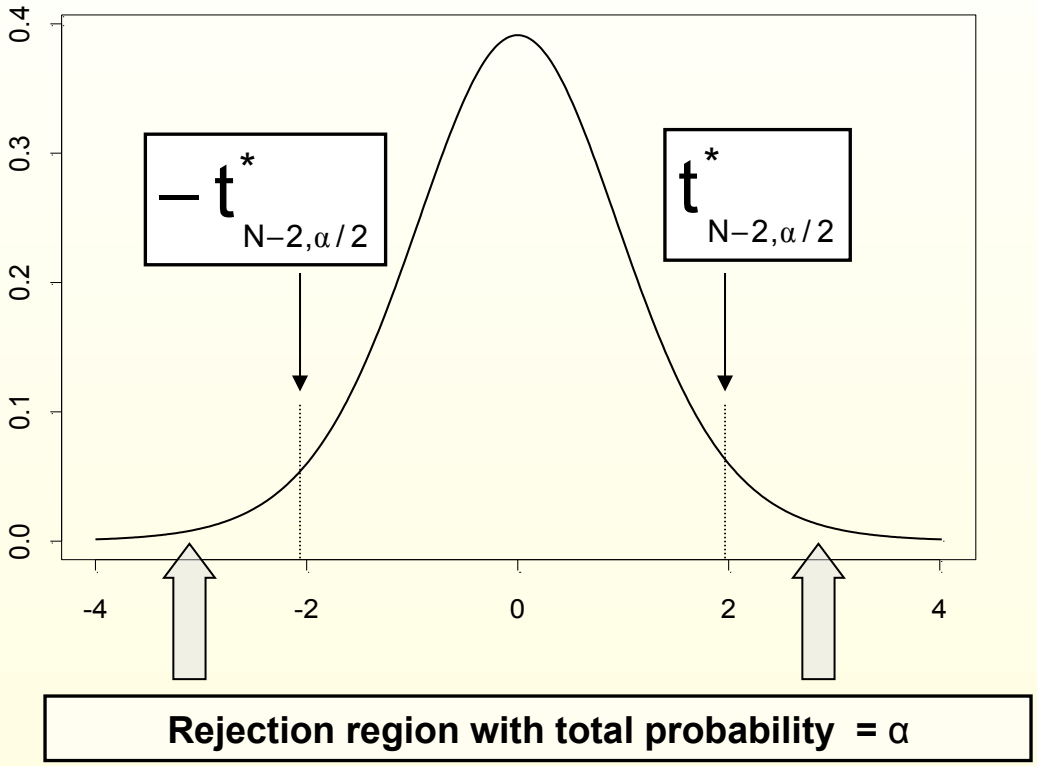
Formal Approach to Hypothesis-Testing:

Two Steps:

i. Pick the **significance level** (α) = Prob(reject when null true) by deciding what level of error of this kind is acceptable (called type I error).

ii. Use α to choose a **rejection region** – the set of t statistic values which will lead to a rejection. This is done by picking a **critical value**, $t^*_{N-2,\alpha/2}$ such that there is α/2 area in the tails to the right and left of t^* and $-t^*$…

j. Hypothesis Testing



The critical value, t^*, can be determined from a "table" of the t-distribution. This is built into R.

j. Hypothesis Testing

In practice, we take a value of α to be around .05 unless:

- Sample size is large (e.g. > 10,000)
- Cost of making a type I error is large

(type I error = reject null when null is true)

In summary, we reject if (otherwise, fail to reject):

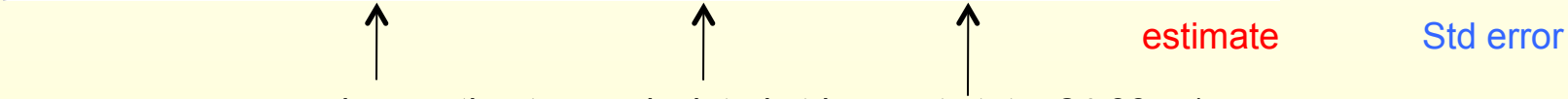
$$|t| = \left| \frac{b_1 - \beta_1^*}{s_{b_1}} \right| > t^*_{N-2,\alpha/2}$$

j. Example: Market Model and Hypothesis Testing

Even though we know it to be false, let's hypothesize that there is no relationship between VGHCX and the Market.

$$H_0: \beta_1 = 0$$
$$H_A: \beta_1 \neq 0$$

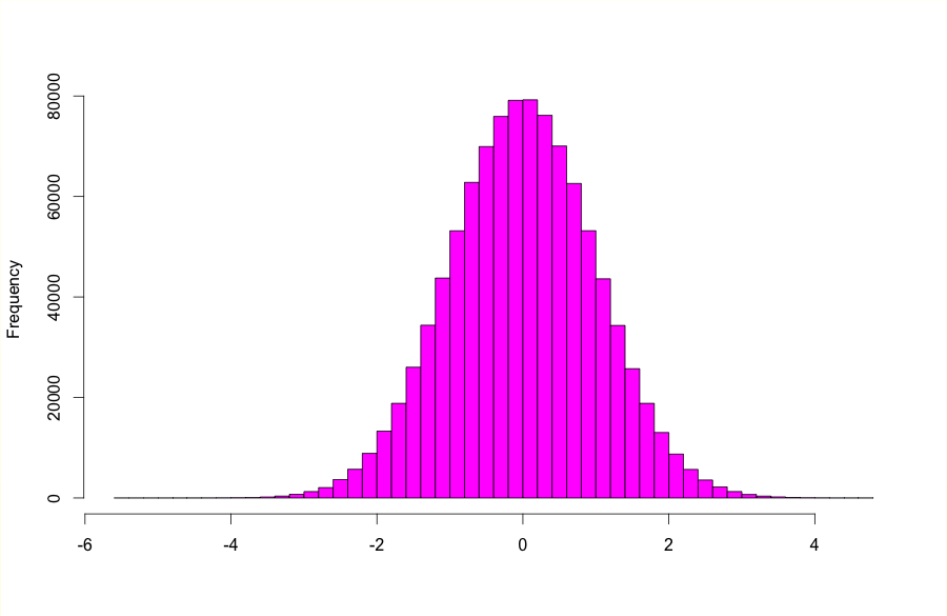
```
Coefficients:
             Estimate Std Error t value p value
(Intercept) 0.006528  0.001374    4.75       0
vwretd      0.743000  0.030130   24.66       0
```



j. Example: Market Model and Hypothesis Testing

The t value is huge (24) relative to the null t distribution with 349-2=347 degrees of freedom!

To illustrate just how big this is, let's simulate some numbers from the t distribution



In 100,000 draws from the t distribution, we didn't get a single value anywhere *near* 24.

We conclude that we reject the null hypothesis: H_0: $\beta_1 = 0$

j. Example: Market Model and Hypothesis Testing

Now let's test a more relevant value of β_1

Stocks and portfolios are characterized by their betas which are estimated from regressions very similar to this one. β_1 is sometimes used as an estimate of risk or volatility. The value of 1 has central significance.

$\beta_1 > 1$: volatile assets (amplify market up/down moves)

$\beta_1 < 1$: non-volatile assets (shrink market movements)

This suggests that we consider the hypothesis that $\beta_1 = 1$.

$H_0: \beta_1 = 1$

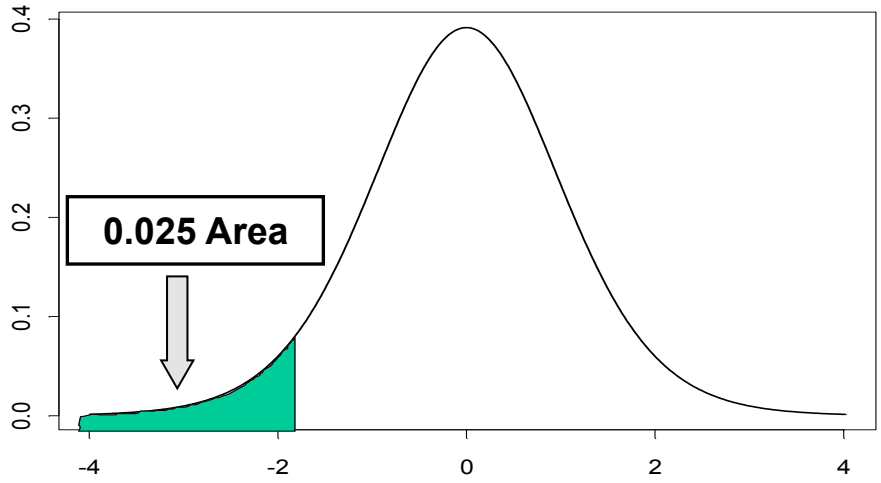
$H_A: \beta_1 \neq 1$

j. Example: Market Model and Hypothesis Testing

Find the critical value, t* for the .05 significance level.

We want the value of t statistic so that:

$Pr[\ |t| > t^*\ ] = .05$



We can use the qnorm (inverse of CDF) command to compute this for us.

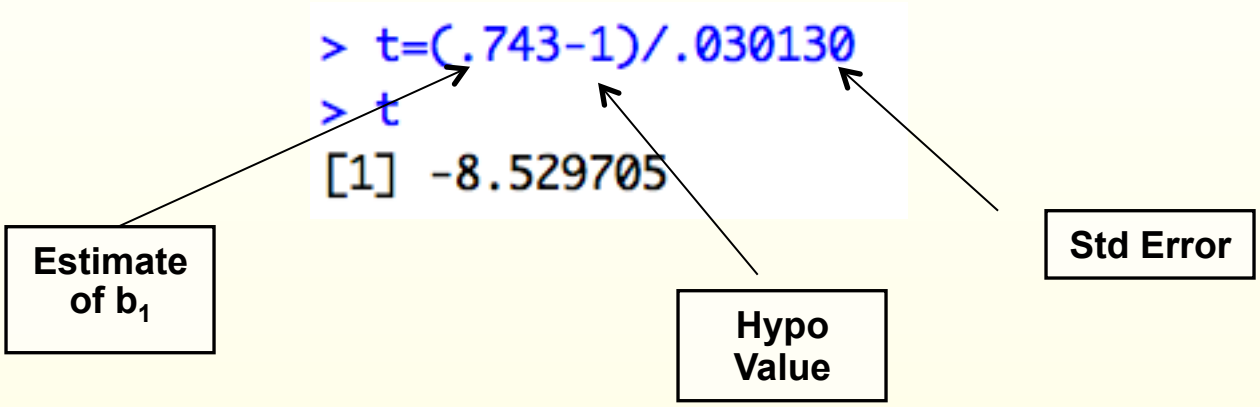
```
> qt(.025,df=349-2)
[1] -1.966824
```

Value such that Pr[t < t*] = 0.025

This is the cut-off value. Notice how close to 1.96 it is

j. Example: Market Model and Hypothesis Testing

Now compute the value of t stat:



8.529 is larger than than the 95% critical value for t(347) so we reject the null hypothesis: H_0: $\beta_1 = 1$

j. Example: Market Model and Hypothesis Testing

Let's look at the intercept for the VGHCX fund regression. Remember this is Jensen's alpha.

$$H_0: \beta_0 = 0$$
$$H_A: \beta_0 \neq 0$$

```
Coefficients:
             Estimate Std Error t value p value
(Intercept) 0.006528  0.001374    4.75       0
vwretd      0.743000  0.030130   24.66       0
```

We can see that the intercept is significantly different from zero. However, 95 CI is large:

[.0038, .0092]

j. P Values

One of the problems with formal hypothesis-testing is that the strength of information in the data in support/against null is not conveyed by accept/reject!

- t value is a tiny bit less than the t cutoff, we accept the null
- If the t value is a tiny bit bigger we reject the null

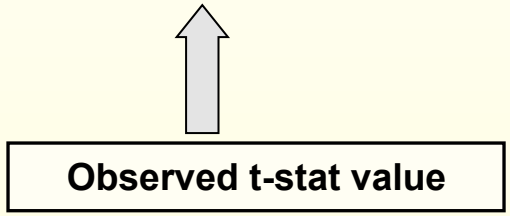
The information from the data is pretty much the same but we act quite differently.

Therefore, we need some measure of the strength of rejection. The value of the t stat itself is not a good candidate for this since we want to gauge how unusual this value is relative to the t statistic null distribution. The **p-value** is designed to do this.

j. P Values

The p-value is the probability of observing a value of the t statistic farther out in the tail than the observed t value.

$$p = \Pr\left[\,|t_{N-2}| \geq |t|\,\right]$$



For the standard t-tests printed out by R, the p-value is computed automatically. However, for non-standard tests, we need to compute the p-value. Here we need a CDF (value to probability) table or function.

j. P Values

Let's compute *p* for the mutual funds example of testing $\beta_1 = 1$. `pt()` is the R function for the CDF of a t distribution.

```
> t=(.743-1)/.030130
> pvalue=2*pt(-abs(t),df=347)
> pvalue
[1] 4.607926e-16
```

Another way of using the p-value FOR ANY test statistic:

The p-value is the minimum significance level at which you can reject the null.

e.g. p = .03 -> reject at .05 but not at .01 level.

P values and Testing Summarized

Small p value ($< \alpha$) $\Longleftrightarrow$ large $|t|$ $\Longrightarrow$ reject

Large p value ($\geq \alpha$) $\Longleftrightarrow$ small $|t|$ $\Longrightarrow$ accept null

k. Prediction

Model: $Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \quad i = 1, \ldots, N$

$$\varepsilon_i \sim \text{iid} N\left(0, \sigma^2\right)$$

The *conditional forecasting problem* can be succinctly stated as:

- Predict a "future" observation, y_f
- Given X_f and the sample data $\{X_i , Y_i \}$ i = 1, …, N

The only practical solution to the prediction problem is to use estimated parameters:

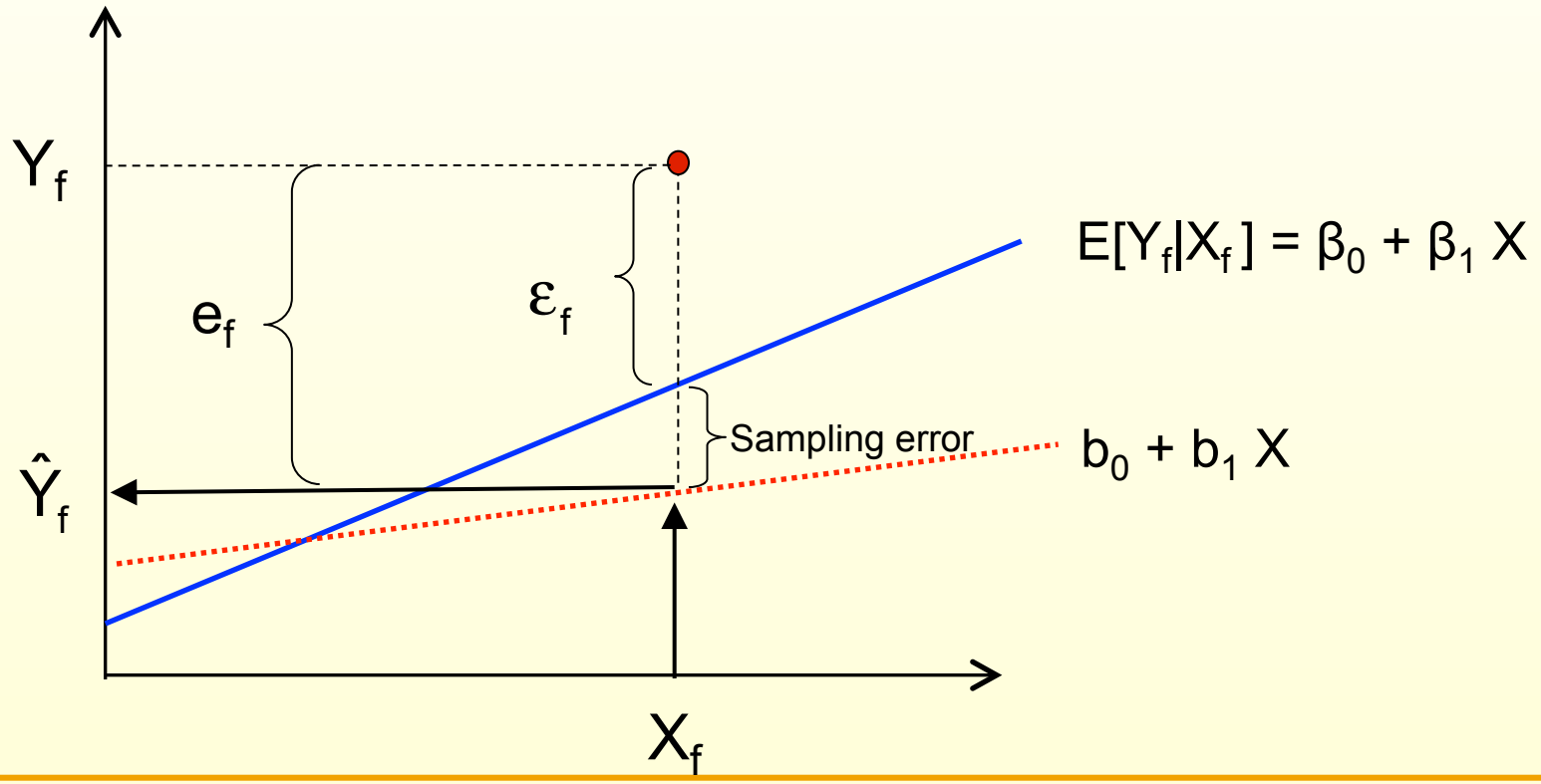
$$\hat{Y}_f = b_0 + b_1 X_f$$

k. Prediction

If we use this predictor, we will make a *prediction error*:

$$e_f = Y_f - \hat{Y}_f = Y_f - b_0 - b_1 X_f$$

Let's draw this:



k. Prediction

Let's write our prediction error in such a way so that we can see the influence of two factors:

i. the model error term or the inherent randomness

ii. estimation error in the model parameters (sampling error)

$$Y_f - \hat{Y}_f = e_f = Y_f - E[Y_f \mid X_f] - \left(\hat{Y}_f - E[Y_f \mid X_f]\right)$$

$$= \left[\beta_0 + \beta_1 X_f + \varepsilon_f - (\beta_0 + \beta_1 X_f)\right] - \left[\hat{Y}_f - E[Y_f \mid X_f]\right]$$

$$= \varepsilon_f - \left[\hat{Y}_f - E[Y_f \mid X_f]\right]$$

Sampling Error

$$= \varepsilon_f - \left[(b_0 - \beta_0) + (b_1 - \beta_1) X_f\right]$$

Inherent Randomness

k. Prediction

Now let's compute a prediction interval for Y_f

$$\mathrm{Var}\left(e_f = Y_f - \hat{Y}_f\right) = \mathrm{Var}\left(\varepsilon_f\right) + \mathrm{Var}\left(\hat{Y}_f\right)$$

$$= \sigma^2 + \sigma^2\left(\frac{1}{N} + \frac{\left(X_f - \bar{X}\right)^2}{\left(N-1\right)s_X^2}\right) = \sigma^2\left(1 + \frac{1}{N} + \frac{\left(X_f - \bar{X}\right)^2}{\left(N-1\right)s_X^2}\right)$$

The *predictive standard error*, denoted s_{pred}, is then

$$s_{pred} = s\left(1 + \frac{1}{N} + \frac{\left(X_f - \bar{X}\right)^2}{\left(N-1\right)s_X^2}\right)^{.5}$$

Standard Error of the Regression

k. Prediction

Let's return to the printout and fill-in the formula for the prediction interval

```
> predict(out,new=data.frame(vwretd=.10),int="prediction")
         fit        lwr       upr
1 0.08082871 0.03120471 0.1304527
```

$$b_0 + b_1 X_f \pm t^*_{N-2,\alpha/2}\, s\left(1 + \frac{1}{N} + \frac{\left(X_f - \bar{X}\right)^2}{(N-1)s_X^2}\right)^{1/2} = b_0 + b_1 X_f \pm t^*_{N-2,\alpha/2} s_{pred}$$

where $X_f = .10$

Both estimation and prediction problems are solved by proposing an estimator or a predictor. Estimators or predictors are functions of the sample data. For example,

$$\hat{\beta}_1 = b_1 = \frac{\sum (x_i - \bar{x}) Y_i}{\sum (x_i - \bar{x})^2} = \sum c_i Y_i$$

$$\hat{Y}_f = b_0 + b_1 X_f$$

We has discussed the *sampling properties* of estimators or predictors.

In the case of linear regression both the estimator of the coefficients and the predictors are unbiased, i.e.

$$E\left[b_1\right] = E\left[\sum c_i Y_i\right] = \sum c_i E\left[Y_i\right] = \beta_1$$

and

$$E\left[\hat{Y}_f\right] = E\left[b_0 + b_1 X_f\right]$$

$$= E\left[b_0\right] + E\left[b_1\right] X_f = \beta_0 + \beta_f X_f = E\left[Y_f \mid X = X_f\right]$$

In general, particularly for judgmental forecasting or non-linear models, we might not have an unbiased estimator/predictor. How do we measure performance?

If we consider the Mean Squared Error (MSE) metric, then we measure performance by how "close" the estimator/predictor is to the true value.

$$\mathrm{MSE}\left(\hat{\theta}\right) = \mathrm{E}\left[\left(\hat{\theta}-\theta\right)^2\right] = \mathrm{E}\left[\left(\hat{\theta}-\mathrm{E}\left[\hat{\theta}\right]\right)^2\right] + \left(\mathrm{E}\left[\hat{\theta}\right]-\theta\right)^2$$
$$= \mathrm{Variance} + \mathrm{Bias}^2$$

Obviously, for an unbiased estimator/predictor the Bias term is zero. But there can be situations in which biased estimators/predictors can out perform (on the basis of MSE) unbiased estimators/predictors by trading off bias for variance. More later.

MSE is measured in the units of the parameter or the dependent variable squared. We usually consider the square root of MSE, RMSE.

If we have a "track" record of S forecasts, then we can define a sample version of these metrics.

$$\text{sample RMSE} = \sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\hat{Y}_{f,s} - Y_{f,s}\right)^2}$$

Some prefer the absolute value "scoring" or "loss" function, Mean Absolute Error (MAE).

$$\text{MAE}\left(\hat{\theta}\right) = \text{E}\left[\left|\hat{\theta} - \theta\right|\right]$$

Appendix: Derivation of LS Slope

We have satisfied ourselves that corr(e, X) =0. Now let us use this intuition to derive the least squares formula for b_1

Verification:

realize that corr(e,X) = 0 is equivalent to cov(e, X) = 0 (why?)

$$\text{cov}(e,X) = \text{cov}(X,e) = 1/(N-1)\sum\left(X_i - \bar{X}\right)(e_i)$$

substitute in for e:

$$\sum\left(X_i - \bar{X}\right)\left(Y_i - b_0 - b_1 X_i\right) = \sum\left(X_i - \bar{X}\right)\left(Y_i - \left[\bar{Y} - b_1\bar{X}\right] - b_1 X_i\right) = 0$$

$$b_0 = \bar{Y} - b_1\bar{X}$$

Appendix: Derivation of LS Slope

Verification (continued):

$$\sum\left(X_i-\bar{X}\right)\left(Y_i-\left[\bar{Y}-b_1\bar{X}\right]-b_1X_i\right)=0$$

or

$$\sum\left(X_i-\bar{X}\right)\left(Y_i-\bar{Y}-b_1\left(X_i-\bar{X}\right)\right)=0$$

or

$$\sum\left[\left(X_i-\bar{X}\right)\left(Y_i-\bar{Y}\right)-b_1\left(X_i-\bar{X}\right)^2\right]=0$$

Solving for b_1

$$b_1=\frac{\sum(X_i-\bar{X})(Y_i-\bar{Y})}{\sum(X_i-\bar{X})^2}$$

Glossary of Symbols

X= independent or explanatory variable

Y= dependent variable

b_0 - least squares estimate of intercept

b_1 - least squares estimate of regression slope

e_i - least squares residual

$\hat{Y}$ - fitted value

s - sample standard dev (subscript tells you of what var)

s^2 - sample variance (subscript tells you of what var)

r - sample correlation coefficient

SST - total sum of squares

SSR - regression sum of squares

SSE - error sum of squares

R^2 - coefficient of determination, goodness of fit measure

Glossary of Symbols

Regression Model parameters

β_0 - true line intercept

β_1 - true slope

σ or σ_ε - error standard deviation

ε - true regression line errors or model errors

Glossary of Symbols

X_f - future value of X for forecasting

Y_f - value of Y to be forecasted

ε_f - prediction error when forecasting from true line

e_f - prediction error when forecasting from fitted line

r_i - standardized residual

Important Equations

$$\hat{Y}_i = b_0 + b_1 X_i$$

fitted value

$$e_i = Y_i - \hat{Y}_i$$

residual

$$b_1 = \frac{\sum_{i=1}^{N}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{N}(X_i - \bar{X})^2} = \frac{s_{XY}}{s_X^2}$$

least squares formulae for slope and intercept

$$b_0 = \bar{Y} - b_1 \bar{X}$$

Important Equations

$$\hat{Y} - \overline{Y} = b_1(X - \overline{X})$$

least squares line passes thru point of means

$$b_1 = \frac{s_{xy}}{s_x^2} = r_{xy} \times \frac{s_y}{s_x}$$

relationship between slope coef and correlation

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

Alternative definitions for R-squared

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$
$$\varepsilon_i \sim \text{iid } N(0, \sigma^2)$$

$$Y \mid X = x \sim N\left(\beta_0 + \beta_1 x,\ \sigma^2_{Y|X}\right)$$

Two Versions of Simple Linear Regression Model:

top as regression equation

bottom as conditional distribution

Important Equations

$$s^2 = \frac{1}{N-2}\sum_{i=1}^{N} e_i^2 = \frac{SSE}{N-2}$$

estimate of error variance

$$s = \sqrt{\frac{SSE}{N-2}}$$

standard error of the regression

$$Var(b_1) = \frac{\sigma^2}{\sum_{i=1}^{N}(X_i - \bar{X})^2} = \frac{\sigma^2}{(N-1)s_X^2}$$

three factors driving sampling variance of slope

Important Equations

$$s_{b_1} = \sqrt{\frac{s^2}{(N-1)s_X^2}}$$

std errors of coefs

$$(1-\alpha)\%\ \text{C.I.:}\quad b \pm t_{N-2,\alpha/2} s_b$$

Confidence Interval

$$|t| = \left| \frac{b_1 - \beta_1^*}{s_{b_1}} \right| > t^*_{N-2,\alpha/2}$$

Rejection Region for t-test

$$p = \Pr\left[\,|t_{N-2}| \geq |t|\,\right]$$

definition of p-value

Important Equations

$$e_f = Y_f - \hat{Y}_f = Y_f - b_0 - b_1 X_f$$

definition of prediction error

$$\mathrm{Var}\left(e_f = Y_f - \hat{Y}_f\right) = \mathrm{Var}\left(\varepsilon_f\right) + \mathrm{Var}\left(\hat{Y}_f\right)$$

decomposition of prediction error

$$s_{pred} = s\left(1 + \frac{1}{N} + \frac{\left(X_f - \bar{X}\right)^2}{\left(N-1\right)s_X^2}\right)^{.5}$$

predictive standard error

Glossary of R commands

- **`abline(c(intercept,slope)):`** Adds one line through the current plot with the defined intercept and slope
- **`anova():`** Computes analysis of variance (or deviance) tables for one or more fitted model objects.
- **`c()`**: Combine values into a vector or list
- **`cov(x,y),cor(x,y):`** Compute the covariance or correlation of variables x and y.
- **`data(A)`** : allows access to data frame A that is in the data library.
- **`head(A)`** : Returns the first parts of a data frame A.
- **`library("PERregress")`** : loads the R package containing datasets and customized functions for our class.
- **`lm(Y~X):`** Fits a linear model. Y is dep var and X is indep var.

Glossary of R commands

- **`plot(X,Y):`** Plots X and Y
- **`quantile(x,probs=c()):`** computes the quantiles of x
- **`predict()`**: Predictions from the results of various model fitting functions.
- **`str(object)`**: tells you the "structure" of an object, e.g. str(dataframe).
- **`summary():`** Generic function used to produce result summaries of the results of various model fitting functions.
- **`var(X),mean(X),sd(X)`**: Compute the variance, mean, standard deviation of a variable named X.
- **`sum(x):`** computes the sum of elements in x.

Glossary of R commands

- **`descStat(A):`** produce descriptive statistics for all the variables in data frame A that are not factors.
- **`cut(x,breaks=c())`**: "cut" continuous variable x into intervals defined by the "breaks" vector. Produces a factor or categorical variable.
- **`merge(df1,df2,by="var_name"):`** merge df2 into df1 based on values specified by the "by" variable
- **`hist(a):`** Graphs a histogram of the given data values of the variable a.
- **`pf(t stats,df=10)`**: Returns the p-value of a t statistic with degrees of freedom = 10
- **`qt(prob,df=10)`**: Returns the t statistics for the left-tail probability of a t distribution with degree of freedom = 10.
- **`rt(100,df=10):`** Generates random 100 numbers for the t distribution with degree of freedom of 10.

- **`normPlot(variable name)`**: This is a customized function to return the normal probability plot of a variable.