Education

AFM 113 Final Prep

Disclosure: This material is for educational purposes only and is intended to supplement course content. Please ensure you review the class materials independently.

1.0 Simple Linear Regression

1.1 Linear Regression Manual Process

Simple Linear Regression: Analysis only includes one independent variable and the relationship between the independent (X) and dependent (Y) variables is represented by a straight line.

Exact Model:

- $Y = \alpha + \beta X + \varepsilon$
  - *Alpha* (α) represents the intercept.
  - *Beta* (β) represents the coefficient of the independent variable (the slope).
  - *Epsilon* (ε) represents the random distances from the individual points to the best fitting line (residual terms).
- Goal: Identifying the best fit that will represent the relationship between X and Y in an equation as accurately as possible (measure the best line that fits the data).

Prediction Model:

- $\hat{Y} = \alpha + \beta X$
  - *Y-hat ($\hat{Y}$)* represents the predicted model dependent values.
  - $Y = \hat{Y} + \varepsilon$

Sum of Squared Residuals (Least Squares Method):

$$SSR = \sum_{i=1}^{N} (\alpha + \beta x_i - y_i)^2$$

Mean (μ):

$$\mu = (\sum_{i=1}^{N} x_i)/N$$

Standard Deviation (σ):

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \mu)^2}{N}}$$

1.2 Linear Regression R Code

```
summary(lm(dependent_variablename ~ independent_variablename, dataset))
```

^ Results will reveal the intercept (α) & the slope/coefficient (β).

2.0 Probability Density Curve

2.1 Probability Density Curve Manual Process

Probability Density Curve: Visualizes the probability distribution, allows us to see how probabilities are distributed over the values of a random variable.

Features of Density Curves:

- A density curve must lie on or above the horizontal axis.
- Area under the density curve (between curve & horizontal axis) must always equal 1 or 100%.
- Probability density will always be between 0 & 1 (area under the curve).
- Probability density ≠ probability.
    - Probability density compares the likelihood of observing a value.
    - Probability compares the probability of the value falling in a range of observations.
- For a continuous variable ($-\infty < x < \infty$), discussing its probability of being a specific value is not meaningful because it always equals 0.
    - Pr(x = #) = 0

Outliers: A data point that significantly deviates from the general pattern or average of the rest of the data points in a dataset.

- Mean > Median ⇒ Right-Skewed (upper)
- Mean < Median ⇒ Left-Skewed (lower)

2.11 General Normal Distribution

$X \sim N(\mu, \sigma^2)$

The normal distribution is characterized by its mean (μ) and/or standard deviation (σ).

- Mean tells us where the center of the curve is.
- Standard deviation tells us how wide the curve is (will also determine the height of the curve).

2.12 Standard Normal Distribution

$Z \sim N(0, 1)$

Properties of the standard normal distribution (Z probabilities):

- If Pr ($-n < Z < n$), then = Pr ($z < n$) -Pr ($z < -n$).
- If Pr ($z < 0$), then = Pr ($z > 0$) = 50%.
- If Pr ($z > a$), then = Pr ($z < -a$).
- If Pr ($z > a$), then = 1-Pr ($z < a$).

Standardization: Transforming a general normal distribution into the standard normal distribution.

- $Z = (X - \mu) / \sigma$

2.2 Percentage Returns R Code

```
Returns <- dataset %>% mutate(dependent_returns = (dependentvariablename - lag(dependentvariablename)) / lag(dependentvariablename), independent_returns = (independentvariablename - lag(independentvariablename)) / lag(independentvariablename)) %>% na.omit()
```

^ Results will give the percentage returns of the variables.

^ lag() ← Captures end-of-previous month prices.

2.3 Beta Rolling Window R Code

```
rollBeta <- data.frame(WindowEndMonth = as.Date(character()), beta = numeric(), stringsAsFactors = FALSE)
for(i in 1:(nrow(dataset) - #-1)) {
subset_df <- dataset[i:(i+#-1), ]
subset_beta <-lm(dependent_variablename ~ independent_variablename, subset_df)
beta_coef <- coef(subset_beta)["independent_variablename"]
rollBeta <- rbind(rollBeta, data.frame(WindowEndMonth = subset_df$datevariablename[#],
beta = beta_coef, row.names=NULL)) }
```

^ Identifies an estimated beta value for a dependent variable using a rolling window cycle (automatic rolling window on R → line 2&3, remove line 2&3 for manual).

2.4 Graph Generation R Code

```
binwidth <- (max(dataset$variablename) - min(dataset$variablename))/#ofbins
bin_edges <-
seq(min(dataset$variablename),max(dataset$variablename),binwidth)
```

^ Determine the cutoff values for histogram bins manually (display range of returns for each bin).

```
Ggplot(dataset, aes(x = variablename, y =..density..)) + geom_point() +
geom_smooth(method = “lm”, se = False) + geom_histogram(breaks = bin_edges,
fill = “color”, alpha = #) + geom_line() + geom_density(color = “color”, size
= #) + geom_qq() + geom_vline(xintercept = #, linetype = "type") +
geom_abline() + scale_axis_continuous(labels = scales::percent) + labs(title
= “title”, x= “dependent_variablename”, y = “independent_variablename”) +
theme(plot.title = element_text(hjust = #), axis.text/title.axis =
element_text(angle = #)
```

^ geom_point() ← scatterplot, geom_smooth() ← smooth line, geom_histogram() ← histogram, geom_line() ← time-series plot, geom_density() ← probability density curve, geom_qq() ← quantile-quantile plot, geom_vline() ← vertical line, geom_abline() ← reference line (intercept = 0, slope = 1).
^ scale_axis_continuous() ← Specifies which axis to scale continuously.
^ theme_classic() ← no grid lines, theme_bw() ← grey grid lines, theme() ← Customized looks.
^ se = False ← without the confidence interval, ..density.. ← scale y-axis from frequency to probability density, scales::percent ← Label the scales in percentage.

2.3 Probability Calculation R Code

```
pnorm(x_value, mean = #, sd = #)
```

^ Results will identify the probability at x-value (z value, mean = 0, sd = 1).
^ dnorm() ← gives probability density, qnorm() ← finds quartile/percentile.

3.0 Normal Distribution

3.1 Normal Distribution Manual Process

If a dataset follows standard normal distribution, then Pr(−1 < Z < 1) determines "the percentage of observation [that] lies within one standard deviation of the mean."

^ *mean (μ) + standard deviation (σ) = 0+1 = 1, mean (μ) - standard deviation (σ) = 0-1 = -1*

- 68% of the data lies within one standard deviation of the mean.
- 95% of the data lies within two standard deviations of the mean.
- 99% of the data lies within three standard deviations of the mean.

3.11 IQR-to-SD ratio

IQR / SD ≈ 1.34

^ No fixed universally applicable ratio (only testing through one method is not enough to identify if a dataset follows a standard normal distribution or not).

3.12 Quantile-Quantile Normality Plot

Pr(Y < y1) = Pr(X < x1) = Pr(Z < z1) = 1/(n+1)

^ Find the probability of the z-score using 1/(n+1) then use standardization to find the x value.

The x values should be very similar with the y values (thus should have a 45° straight line to be a normal distribution).

3.2 Normal Distribution R Code

```
dataset %>% mutate(
range1 = if_else(variablename <= mean(variablename) + sd(variablename) &
variablename >= mean(variablename) - sd(variablename),1,0),
range2 = if_else(variablename <= mean(variablename) + 2*sd(variablename) &
variablename >= mean(variablename) - 2*sd(variablename),1,0),
range3 = if_else(variablename <= mean(variablename) + 3*sd(variablename) &
variablename >= mean(variablename) - 3*sd(variablename),1,0))
```

^ Results will identify the ranges of standard deviations from the mean.

```
dataset %>% summarize(pct_range1 = mean(range1), pct_range2 = mean(range2),
pct_range3 = mean(range3))
```

^ Finds the number of observations that fall within the mean of the ranges.

3.2 QQ Normality Plot R Code

```
dataset <- dataset %>% mutate(rank = rank(variablename), percentile =
rank/(nrow(dataset)+1), stdnorm = qnorm(percentile))
QQ plot (x=stdnorm, y = (variablename - mean(variablename)/sd(variablename)))
```

^ Allows you to create a QQ plot for any distribution.

```
ggplot(dataset, aes(sample = variablename)) + geom_qq() + geom_qq_line(colour
= "colour", linewidth = #) + labs(x = "Theoretical Returns", y = "Sample
Returns")
```

^ Direct QQ plot for sample.

4.0 Sample Probability Distribution

4.1 Sample Probability Distribution Manual Process

Sample Distribution: Inferring the unknown population mean based on the known sample mean.

Standard Error: Standard deviation of the sample (about 10 times smaller than the population standard deviation).

- The more samples we have, the smaller the difference between the sample mean and population mean.

$$\bar{x} \sim \mathcal{N}(\mu_{\bar{x}}, \sigma^2_{\bar{x}})$$

The mean of the sample means equals to the population mean:

$$\mu_{\bar{x}} = \mu$$

The standard error equals the population's standard deviation divided by the square root of the sample size:

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$$

Central Limit Theorem (CLT): By satisfying the following three conditions, the samples will follow a normal distribution, even if the population does not (z-score can be used):

- We must always be able to draw out multiple samples out of the population.
- We must know the mean and standard deviation of the population,
- The sample size must be sufficiently large (greater than or equal to 30 observations), if it is not a normally distributed population.

4.2 Sample Probability Distribution R Code

```
set.seed(#)
sampleset <- dataset %>% slice_sample(n = #)
```

^ Seed value can be any value (links the value to a fixed set of sample observations).

```
dataset <- data.frame(variablename = numeric())
for (i in 1:10) {
set.seed(i)
sample <- dataset %>% slice_sample (n = #)
temp_var <- mean(sample$variablename)
mvar <- rbind(mvar, data.frame(avg_var = temp_var, row.names = NULL))}
```

^ Results allows for estimation of the variable's average variability (taking random samples & finding the mean of each, compiling it all together to compare).

5.0 Confidence & Significance Levels

5.1 Confidence & Significance Levels Manual Testing

Sample Standard Deviation: Where s = sample standard deviation, n = # of observations, and $\bar{x}$ = sample mean.

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}$$

Point Estimator: Claim of a specific, singular point.
Interval Estimator: Claim of a range that a point may fall into (more confident).

The confidence interval (CI) for an unknown population mean can be written as:

$$\bar{X} - \text{MoE} \leq \mu \leq \bar{X} + \text{MoE}$$

Margin of Error (MoE): Considered a buffer zone.

- MoE = Critical Value * Standard Error

Standard Error: Population (or sample) Standard Deviation / Sqrt of # of Observations

$$SE = \frac{\sigma \ (or \ s)}{\sqrt{n}}$$

Critical Value = $Z_{\frac{\alpha}{2}}$

Confidence Level: Measures how confident we are that the calculated interval contains the true (but unobservable) population mean.

- Most Common Confidence Level: 90% / 95% / 99%.
    - e.g., Pr( ) = 95%
- The greater the confidence level is, the wider the confidence interval is.

5.11 Known SD

Scenario 1: σ is known, confidence level is 95%.

- Pr($\bar{x}L < \bar{x} < \bar{x}U$) = 95% (1 - α)
- This means the left over = 5% (α), 2.5% on each side (α/2) ← two-tailed.
- That means Pr($\bar{x} < \bar{x}L$) = 2.5%
- Standardization of sample to standard: $Z = (\bar{x} - \mu) / (\sigma / \sqrt{n})$
- Generally: Pr(-Z α/2 < Z < Z α/2)
- Therefore when σ is known: $\bar{x}$ - Z α/2 * $(\sigma / \sqrt{n}) \leq \mu \leq \bar{x}$ + Z α/2 * $(\sigma / \sqrt{n})$ = 1 - α

5.12 Unknown SD

Scenario 2: σ is UNKNOWN, confidence level is still 95%.

- Replace the population sd with the sample sd (s).

$$t = \frac{\bar{X} - \mu}{s/\sqrt{n}}$$

- This will no longer give a standard normal distribution (Z). As a result, we use the t-distribution or t-score (t-distribution has fatter tails than the normal distribution).
- By increasing more of the observations, the t-distribution will become more and more like the standard normal distribution.

$$\bar{X} - t_{\alpha/2}\left(\frac{s}{\sqrt{n}}\right) \leq \mu \leq \bar{X} + t_{\alpha/2}\left(\frac{s}{\sqrt{n}}\right)$$

^ Replaced all of Z with t & all of σ with s.

5.2 T-Statistic R Code

```
sample_size <- #
alpha <- #
confidence_level = 1-alpha
t-value <- qt(confidence_level,df= sample_size - 1)
```

^ Identifies the quantiles for t-distribution.
^ Use qnorm() to find z-scores ← check: see 2.3 Probability Calculation R Code.

```
mu <- mean(variablename)
sigma <- sd(variablename)
margin_error <- t_value * (sigma / sqrt(sample_size))
ci_lower <- mu - margin_error
ci_upper <- mu + margin_error
```

^ Finds the MoE and confidence interval of the sample.

6.0 Hypothesis Testing

6.1 Hypothesis Testing Manual Process:

- Null Hypothesis: What we are testing.
- Alternative Hypothesis: Anything but the null hypothesis.
- As words:
  - Null Hypothesis: The mean is equal to ___.
  - Alternative Hypothesis: The alternative hypothesis is not ___(null hypothesis).
- As symbols:
  - H_0: μ = ___
  - H_a: $\mu \neq$ ___
- NOTE: The null hypothesis can NOT be an inequality.
- Rejection region:
  - SD known: $Z < -z_{\alpha/2}$ or $Z > z_{\alpha/2}$
  - SD unknown: $t < -t_{\alpha/2}$ or $t > t_{\alpha/2}$

Two-Tailed Test:

- Alternative hypothesis is non-directional, solely rejecting the null hypothesis.
    - e.g., $H_a : \mu \neq \mu_0$
    - Rejection region is equally split between both tails of the distribution (each area accounts for $\alpha/2$).

One-Tailed Test:

- Alternative hypothesis is directional, in which the entire rejection region (α) will be at one tail of the distribution (lower or upper).
    - e.g., $H_a : \mu < \mu_0$ ($H_a = \mu > \mu_0$)
        - Only one critical value is needed.
        - $Pr(t < t^*) = \alpha$
        - Depending on the H_a, the p-value probability sign will follow the same way.

P value: Calculating critical value(s) and the area beyond them to determine if H_0 is rejected.
R Code: Calculating rejection range using P value, which can be done by:

```
xbar <- 173.02
mu0 <- 175
s <- 10.95
n <- 162

# Calculate t-statistic
t_stat <- (xbar-mu0)/(s/sqrt(n))

# Determine p-value
area_lower <- pt(t_stat, df = n-1)
area_upper <- 1- pt(-t_stat, df = n-1)

#return rejection region
area_lower+area_upper
```

Element	One-Sample	Two-Sample
Population Parameter	μ	$\mu_1 - \mu_2$
Sample Statistic	$\bar{X}$	$\bar{X}_1 - \bar{X}_2$
Standard Error	$\frac{\sigma}{\sqrt{n}}$ or $\frac{s}{\sqrt{n}}$	$\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}$ or $\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$
Degree of Freedom	$n - 1$	$\frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{1}{n_1 - 1}\left(\frac{s_1^2}{n_1}\right)^2 + \frac{1}{n_2 - 1}\left(\frac{s_2^2}{n_2}\right)^2}$