**Pierre Carpentier- Springboard Capstone Project Milestone Report 2**

1. **Recap on Project proposal: what is the problem that I want to solve?**

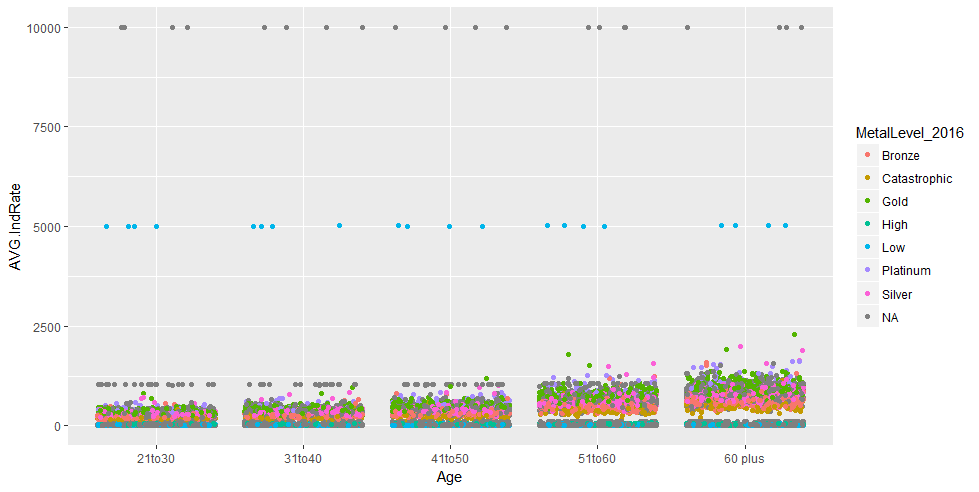
With all the excitement of the US election past us, being able to predict future outcomes in reimbursement rates for private plans in the US healthcare system will become critical for companies’ in that market to adjust their positioning. First, an understanding of what the past 8 years of health care reform has changed in the revenue cycle structure of the market is critical. When the Affordable Care Act was implemented, it created a market of health coverage plans that was subsidized in part by the government, for which any individual can sign up for health coverage at a lower rate. The idea was to expand the grow the existing pool of patients in a free market type of environment in the hopes that the competition would help drive value, in terms of health benefit coverage, to millions of Americans that currently could not afford it. While the Affordable Care Act directly addressed access issues, it did little to address cost issues in terms of pricing for health care services. My thesis is that given pricing for services was not addressed in the Affordable Care Act, and that many common services are priced significantly higher than comparable services in other countries, and given that government subsidies are bridging the gap, the marketplace health plan rates do not differ significantly from private insurance rates pre-Affordable Care Act implementation. This would present a big problem to the newly elected federal government, and limit their options for reforming Health Care law.

1. **A deeper dive into the datasets**

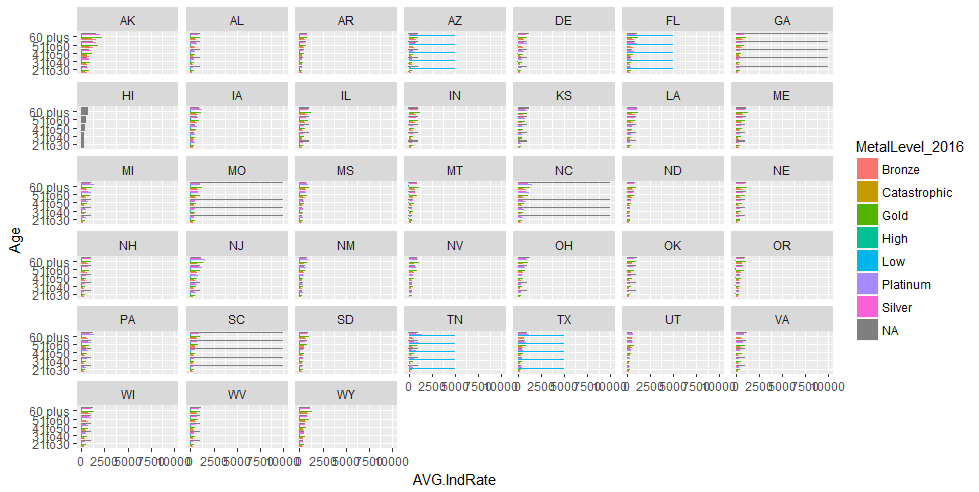
In order to assess how much data wrangling would be required to achieve a useable data set, I started by looking at 7 publicly available data sets for US HealthCare on kaggle. I mapped the various data objects and their meaning and managed to find an exploratory guide published by the Center for Medicare and Medicaid Services. The 7 data sets and a broad description of the important information are included below:

1. BenefitsCost Sharing: Includes very detailed information about each Plan’s inclusions and exclusions, as well as each service type’s impact on co-insurance and co-pay rates. It was quickly concluded that the information in this data set is too granular to begin a high-level analysis there I did not use it for the first milestone.
2. Business Rules: Includes information about the max numbers of dependants and ages for each family Plan as well as some information about Dental coverage only plans. While the overall goal is to provide insight on the value of each plan independently, it was decided to focus on independent coverages so as not to overcomplicate the initial model
3. Network: Includes information about Network commercial plan names and URL information for these plans. I intend to use this data set if any outliers are uncovered while analyzing the various groups of plans.
4. Plan Attributes: Includes information about Max Out of Pocket and Deductible amounts for each plan. After some series of data wrangling, which can be seen in the Capstone Script code, the data frame Attributes3 containing 7639 observations of 4 key variables was produced. These 4 key variables were selected to focus on obtaining maximum financial exposure for individual Plans. The Attributes3 data frame also contains a data object ‘AVCalculatorOutputNumber’ which seems to have a relation to max out of pocket expenses. After further research, it turns out that this variable is the ‘Actuarial Value’ of the plans as calculated by CMS using a complicated metric. Any plans below 0.7 are Bronze, from 0.7 to 0.8 are Silver, from 0.8 to 0.9 are Gold, and 0.9+ are Platinum. Understanding this variable was critical in ensuring that the maximum amount of plans were given the proper Metal level.
5. Rate: By far the largest data set, it includes information about different plans and their various rates or premiums. Correctly sorting the information this data set is critical to being about to derive insight from the various insurance plan policies. To avoid cross-walk confusion, it was decided to filter the data in this set to the Business Year 2016 only. Not only were many plans from 2015 and 2014 observed to be similar to the ones in 2016, but after filtering, we were left with 4.1M observations, which is plenty for the purposes of this exercise. The code is written such that there as 5 ‘Rate’ data frames, therefore if ever it is found that we are required to go back in time to the years 2014-2015 to compare plan values, it is possible to do so. The code included in the capstone folder shows the various iterations to get to Rate5, which is a data frame of 4.1M observations over 17 key variables from the different data sets discussed above.
6. CrossWalk2016: There is various information about how to compare 2015 to 2016 plans in this data set, however, for this Capstone project, it was only used to get the various Plans ‘metal color’ ie: bronze, gold, silver, etc… This information was then merged with the Rate files, which will be the primary data frame on which the remainder of the project will focus.
7. **Using graphics tools to derive insights**

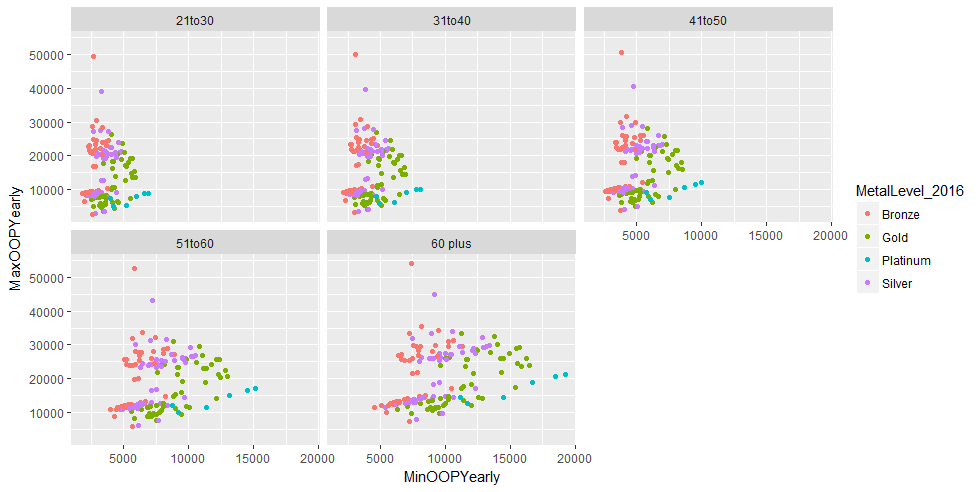
Using basic data visualizations, it became evident that there is a lot of missing data in order to achieve the full scope of the mandate above. Hence, an updated project mandate will be discussed below. First, I manipulated the data in order to get graphical representation of different plan Metal levels and Age buckets using a jitter plot. As expected, there is a slightly upwards trend in average Independent plan rates as you move up in the age buckets. What wasn’t expected are the high numbers for plans with Metal levels ‘NA’ and ‘Low. After further research, it was concluded that these plans are either special plans related to specific issues, or plans that have been grandfathered in. As such, I will likely discard these plans moving forward. IT should also be noted that the plot below is crowded, but for the purpose of general observation, it was adequate for my needs. Should I use this plot in a final report, I will adjust the characteristics to make it more visual.



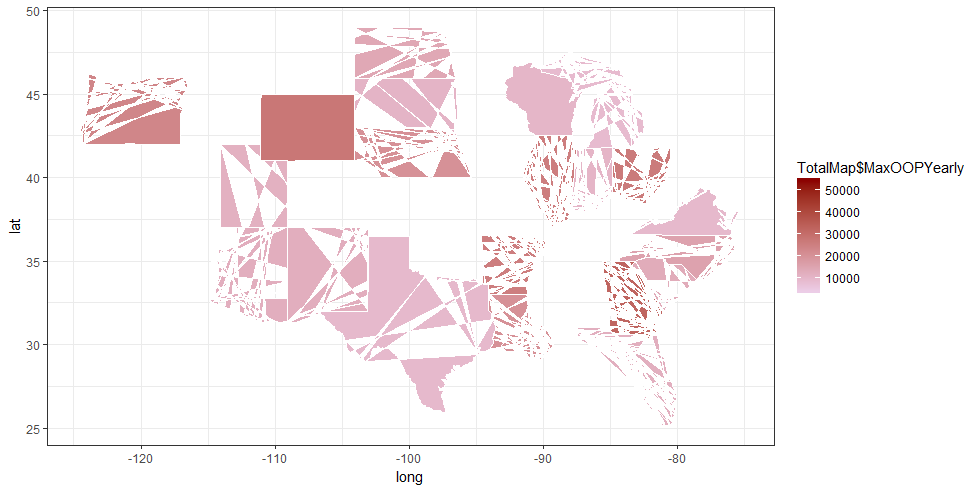
I then tried to use facets to split the data by state, it yielded the graph below. There is again a lot of data on this chart, but viewing it on a large screen yields some interesting insights. States such as Texas, Tennessee, Arizona and Florida have grandfathered some plans in the ‘Low’ Metal level that have very high Independent rates. These states are also known in the industry to have the most predatory insurance products for which the Affordable Care Act is supposed to eliminate. The other States all seem to have plan rates that increase with Metal Level, and increase with Age bucket, which makes sense intuitively. A regression formula might be useful to predict the total yearly spend from a customer’s perspective.



Given the results above, I then calculated 2 metrics: the Maximum yearly out of pocket expense for a customer, and the Minimum yearly out of pocket expense. Plotting the 2 on the x and y axis gives a sense of ‘risk’ adjustment or ‘value’ that a customer is getting for purchasing a plan. Using facets by age group yields interesting insights as well. You can see the 60 plus group has a more spread out grouping and this makes sense because an older population is expected to be riskier for the health system. The plans in the top left of each facet seem to be outliers that allow for large Maximum out of pocket expenses. As expected, the Bronze plans typically offer the lowest minimum payment, but have the largest exposure to max spending.



I then tried to plot some key metrics onto a map using longitude and latitude coordinates, however, it became evident that there are some states missing when you filter the data to focus on individual plans as I did for this project, however, the map does seem to show that the State of Wyoming seems to have the highest yearly maximum out of pocket expenses for health services in the US. I did not bother fixing the polygons as I will likely only use a map in the final report.



All the code to produce these graphs, as well the all the data wrangling, can be found below, as well as on github:

<https://github.com/greedo18/Springboard-Data-Science-Course-Capstone>

1. **Conclusion:**

My final conclusion after data wrangling and exploration is that I will change the mandate of this Capstone project to finding a prediction algorithm that takes into account a patient’s Age, State, and maximum yearly spend on healthcare products to suggest a Plan Metal Level that will be appropriate for them to chose.

1. **Script so far:**

# December 2016- Springboard.com Capstone script for Pierre Carpentier

# Fundamentals of Data Science

# load kaggle data sets for US healthcare policies.

library(readr)

BenefitsCostSharing <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/health-insurance-marketplace-release-2016-01-20-15-52-37/health-insurance-marketplace/BenefitsCostSharing.csv")

BusinessRules <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/health-insurance-marketplace-release-2016-01-20-15-52-37/health-insurance-marketplace/BusinessRules.csv")

Network <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/health-insurance-marketplace-release-2016-01-20-15-52-37/health-insurance-marketplace/Network.csv")

PlanAttributes <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/health-insurance-marketplace-release-2016-01-20-15-52-37/health-insurance-marketplace/PlanAttributes.csv")

Rate <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/health-insurance-marketplace-release-2016-01-20-15-52-37/health-insurance-marketplace/Rate.csv")

ServiceArea <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/health-insurance-marketplace-release-2016-01-20-15-52-37/health-insurance-marketplace/ServiceArea.csv")

Crosswalk2016 <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/health-insurance-marketplace-release-2016-01-20-15-52-37/health-insurance-marketplace/Crosswalk2016.csv")

# Exploratory merging of the data

# Take plans from current business Year as the other years are expired plans and no longer available

library(dplyr)

library(tidyr)

Rate1 <- Rate %>% filter(BusinessYear == "2016") %>% arrange(PlanId)

# Keep only interesting columns and sort Age by groups of 10 year buckets

Rate1 <- data.frame(Rate1$StateCode, Rate1$IssuerId, Rate1$PlanId, Rate1$RatingAreaId, Rate1$Tobacco, Rate1$Age, Rate1$IndividualRate)

Rate1 <- arrange(Rate1, Rate1.Age)

Rate2 <- Rate1 %>% filter(Rate1.Age != "0-20")

Rate2$Rate1.Age <- as.numeric(as.character(Rate2$Rate1.Age))

#Expand age buckets using the mutate function

Rate3 <- Rate2 %>% mutate("21-30" = between(Rate1.Age, 21, 30)) %>% mutate("31-40" = between(Rate1.Age, 31, 40)) %>% mutate ("41-50" = between(Rate1.Age, 41, 50)) %>% mutate("51-60" = between(Rate1.Age, 51, 60)) %>% mutate("60+" = Rate1.Age > 60)

#Now to add basic benefits data to the Rates

Benefits1 <- BenefitsCostSharing %>% filter(BusinessYear == 2016)

Attributes1 <- PlanAttributes %>% filter(BusinessYear == 2016 & DentalOnlyPlan == "No")

MetalLevel <- Crosswalk2016 %>% group\_by(PlanID\_2016, MetalLevel\_2016) %>% summarise()

names(MetalLevel)[1] <- paste("Rate1.PlanId")

Rate4 <- left\_join(Rate3, MetalLevel, by = "Rate1.PlanId")

# Get Plan attributes in order to setup cost calculation

Attributes2 <- Attributes1 %>% group\_by(HIOSProductId, AVCalculatorOutputNumber, TEHBCombInnOonIndividualMOOP, TEHBDedCombInnOonIndividual, BenefitPackageId, FormularyId) %>% summarise()

names(Attributes2)[1] <- paste("Rate1.PlanIdshort")

Rate4 <- Rate4 %>% mutate(Rate1.PlanIdshort = strtrim(Rate4$Rate1.PlanId, 10))

# Split Rates by age buckets and Metal Level to setup yearly max spending for individuals

Rate21to30 <- Rate4 %>% filter(Rate4[8] == TRUE) %>% group\_by(Rate1.PlanIdshort, MetalLevel\_2016) %>% mutate(AVG.IndRate = mean(Rate1.IndividualRate)) %>% arrange(Rate1.PlanId)

Rate31to40 <- Rate4 %>% filter(Rate4[9] == TRUE) %>% group\_by(Rate1.PlanIdshort, MetalLevel\_2016) %>% mutate(AVG.IndRate = mean(Rate1.IndividualRate)) %>% arrange(Rate1.PlanId)

Rate41to50 <- Rate4 %>% filter(Rate4[10] == TRUE) %>% group\_by(Rate1.PlanIdshort, MetalLevel\_2016) %>% mutate(AVG.IndRate = mean(Rate1.IndividualRate)) %>% arrange(Rate1.PlanId)

Rate51to60 <- Rate4 %>% filter(Rate4[11] == TRUE) %>% group\_by(Rate1.PlanIdshort, MetalLevel\_2016) %>% mutate(AVG.IndRate = mean(Rate1.IndividualRate)) %>% arrange(Rate1.PlanId)

Rate60plus <- Rate4 %>% filter(Rate4[12] == TRUE) %>% group\_by(Rate1.PlanIdshort, MetalLevel\_2016) %>% mutate(AVG.IndRate = mean(Rate1.IndividualRate)) %>% arrange(Rate1.PlanId)

#Bind them together to get the final Individual RAte file

RateFinal <- bind\_rows(Rate21to30, Rate31to40, Rate41to50, Rate51to60, Rate60plus)

# Group by Short ID and Metal LEvel to get short table versions of the data by age group

Rate21to30G <- Rate21to30 %>% group\_by(Rate1.StateCode, Rate1.PlanIdshort, MetalLevel\_2016, AVG.IndRate) %>% summarise() %>% arrange(Rate1.PlanIdshort)

Rate21to30G$Age <- "21to30"

Rate31to40G <- Rate31to40 %>% group\_by(Rate1.StateCode, Rate1.PlanIdshort, MetalLevel\_2016, AVG.IndRate) %>% summarise() %>% arrange(Rate1.PlanIdshort)

Rate31to40G$Age <- "31to40"

Rate41to50G <- Rate41to50 %>% group\_by(Rate1.StateCode, Rate1.PlanIdshort, MetalLevel\_2016, AVG.IndRate) %>% summarise() %>% arrange(Rate1.PlanIdshort)

Rate41to50G$Age <- "41to50"

Rate51to60G <- Rate51to60 %>% group\_by(Rate1.StateCode, Rate1.PlanIdshort, MetalLevel\_2016, AVG.IndRate) %>% summarise() %>% arrange(Rate1.PlanIdshort)

Rate51to60G$Age <- "51to60"

Rate60plusG <- Rate60plus %>% group\_by(Rate1.StateCode, Rate1.PlanIdshort, MetalLevel\_2016, AVG.IndRate) %>% summarise() %>% arrange(Rate1.PlanIdshort)

Rate60plusG$Age <- "60 plus"

#Bind them together to get shortened version for data exploration

RateFinalG <- bind\_rows(Rate21to30G, Rate31to40G, Rate41to50G, Rate51to60G, Rate60plusG)

#Now to do some visualizations

library(ggplot2)

ggplot(RateFinalG, aes(x = Age, y = AVG.IndRate, col = MetalLevel\_2016)) + geom\_point() + facet\_wrap(~Rate1.StateCode)

ggplot(RateFinalG, aes(x = Age, y = AVG.IndRate, col = MetalLevel\_2016)) + geom\_jitter()

ggplot(RateFinalG, aes(x = Age, y = AVG.IndRate, fill = MetalLevel\_2016)) + geom\_bar(position = "dodge", stat = "identity") + coord\_flip()

ggplot(RateFinalG, aes(x = Age, y = AVG.IndRate, fill = MetalLevel\_2016)) + geom\_bar(position = "dodge", stat = "identity") + coord\_flip() + facet\_wrap(~Rate1.StateCode)

#Need to add more attributes plus remove NA's and group Low, High and Catastrophic, and Bronze, Silver, GOld, Platinum together

#Note: Performed Data manipualtion for Attributes 2 in excel, explained in final report

Attributes2 <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/Attributes2.csv",

+ col\_types = cols(TEHBCombInnOonIndividualMOOP = col\_number(),

+ TEHBDedCombInnOonIndividual = col\_number()))

Attributes3 <- Attributes2 %>% group\_by(Rate1.PlanIdshort, MetalLevel\_2016) %>% summarise(AVG.AV = mean(AVCalculatorOutputNumber), AVG.MOOP = mean(TEHBCombInnOonIndividualMOOP))

RateFinaljoin <- left\_join(RateFinalG, Attributes3, by = c("Rate1.PlanIdshort", "MetalLevel\_2016"))

#Note: Perform some data manipulation in Excel, explained in final report

RateFinaljoin <- read\_csv("C:/Users/greed/OneDrive/Documents/Springboard/Capstone/RateFinaljoin.csv")

RateFinaljoinfilter <- RateFinaljoin %>% filter(MaxOOPYearly != "NA")

#Some more visual exploration

ggplot(RateFinaljoinfilter, aes(x = MinOOPYearly, y = MaxOOPYearly, col = MetalLevel\_2016)) + geom\_point()

ggplot(RateFinaljoinfilter, aes(x = MinOOPYearly, y = MaxOOPYearly, col = MetalLevel\_2016)) + geom\_point() + facet\_wrap(~Age)

#creating a map

library(mapdata)

library(maps)

RateFinaljoinfilter$region <- state.name[match(RateFinaljoinfilter$Rate1.StateCode, state.abb)]

RateFinaljoinfilter$region <- tolower(RateFinaljoinfilter$region)

TotalMap <- merge(states, RateFinaljoinfilter, by = "region")

ggplot() + geom\_polygon(data = TotalMap, aes(x=long, y=lat, group=group, fill=TotalMap$MaxOOPYearly), colour = "white") + scale\_fill\_continuous(low="thistle2",high="darkred", guide="colorbar") + theme\_bw()