# STAT 5385: Lab 2

Willliam Ofosu Agyapong

16/02/2022

## 0.1 Reading in the SENIC dataset

```
# Import the dataset from local drive
senic <- read.table(file = "../Data Sets/Appendix C Data Sets/APPENC01.txt",
                    header = FALSE)

# View first few observations; Everything looks good.
head(senic)
```

```
##   V1    V2   V3  V4   V5    V6  V7 V8 V9 V10 V11 V12
## 1  1  7.13 55.7 4.1  9.0  39.6 279  2  4 207 241  60
## 2  2  8.82 58.2 1.6  3.8  51.7  80  2  2  51  52  40
## 3  3  8.34 56.9 2.7  8.1  74.0 107  2  3  82  54  20
## 4  4  8.95 53.7 5.6 18.9 122.8 147  2  4  53 148  40
## 5  5 11.20 56.5 5.7 34.5  88.9 180  2  1 134 151  40
## 6  6  9.76 50.9 5.1 21.9  97.0 150  2  2 147 106  40
```

```
# Rename columns
coded_names <- c("ID", "LOS", "age", "infec_risk", "cul_ratio", "xray_ratio",
                 "beds", "med", "region", "ADC", "nurses", "AFS")
names(senic) <- coded_names
```

## 0.2 2.64: Which $R^2$ is largest?

```
# Create a new dataset for convenient modeling.
senic_nested <- senic %>%
  # Select variables of interest
  select(LOS, Infection_Risk=infec_risk, AFS, Chest_Xray_ratio=xray_ratio) %>%
  # Transform data to long format to aid nesting
  pivot_longer(-LOS, names_to = "predictor", values_to = "pred_value") %>%
  nest(-predictor)

# Fit a linear regression model between LOS and each predictor,
models <- senic_nested %>%
  mutate(model = map(data, ~ lm(LOS ~ pred_value, data = .)))

# Extract desired performance metrics.
mdl_metrics <- models %>%
  mutate(metrics = map(model, glance)) %>%
  unnest(metrics) %>%
  select(predictor, r.squared, sigma) %>%
  mutate(sigma.squared = sigma^2, .keep = "unused") %>%
  arrange(desc(r.squared))

# Display results
```

```
names(mdl_metrics) <- c("Predictor", "$R^2$", "MSE")
kable(mdl_metrics, escape = F, caption = "Performance metrics for models Involving each predictor")
```

Table 1: Performance metrics for models Involving each predictor

| Predictor | $R^2$ | MSE |
|---|---|---|
| Infection_Risk | 0.2846 | 2.638 |
| Chest_Xray_ratio | 0.1463 | 3.147 |
| AFS | 0.1264 | 3.221 |

Looking at the $R^2$ column, it turns out **Infection Risk** accounts for the largest reduction in the variability of the average length of stay.

### 0.2.1 Interval estiamtes for slopes

```
# Extract confidence estimates along other measures for the slopes.
mdl_estimates <- models %>%
  mutate(estimates = map(model, tidy, conf.int = T, conf.level = 0.95)) %>%
  unnest(estimates) %>%
  filter(term == "pred_value") %>%
  rename(slope = estimate) %>%
  mutate(CI.length = conf.high - conf.low) %>%
  select(-c(data, model, term, statistic, p.value))

# Display results
kable(mdl_estimates, escape = F, caption = "Model results for each predictor along with 95 % confidence in
```

Table 2: Model results for each predictor along with 95 % confidence intervals of $\beta_1$

| predictor | slope | std.error | conf.low | conf.high | CI.length |
|---|---|---|---|---|---|
| Infection_Risk | 0.7604 | 0.1144 | 0.5336 | 0.9872 | 0.4536 |
| AFS | 0.0447 | 0.0112 | 0.0226 | 0.0668 | 0.0442 |
| Chest_Xray_ratio | 0.0378 | 0.0087 | 0.0206 | 0.0549 | 0.0343 |

We observe that the regression line for the predictor, infection risk, appears to differ greatly from those of available facilities and services (AFS) and the chest X-ray ratio. This is further accentuated by the plots that follow.

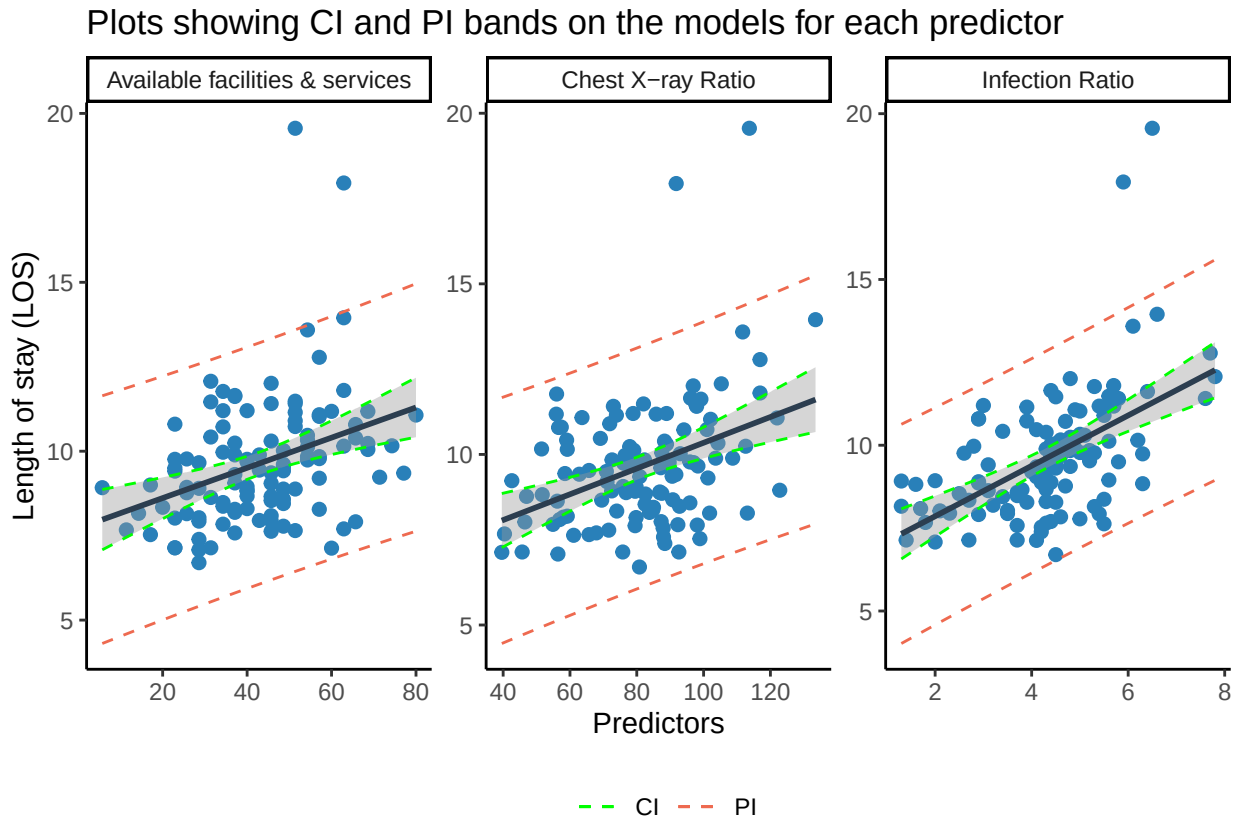
### 0.2.2 CI and PI bands on the models

```
# Append confidence/prediction estimates for the slopes to the data for plotting.
PI_df <- models %>%
  mutate(newdata = map(model, augment, interval = "prediction")) %>%
  unnest(newdata) %>%
  select(PI.lwr = .lower, PI.upr = .upper)

senic_augmented <- models %>%
  mutate(newdata = map(model, augment, interval = "confidence")) %>%
  unnest(newdata) %>%
  bind_cols(PI_df) %>%
  mutate(predictor = recode(predictor,
                            Infection_Risk = "Infection Ratio",
```

```
                          Chest_Xray_ratio = "Chest X-ray Ratio",
                          AFS = "Available facilities & services"
  ))

#----create plot-----
# define colors to help add legend
colors <- c("CI"="green", "PI"="coral2")
ggplot(senic_augmented, aes(x = pred_value, y = LOS)) +
  geom_point(color='#2980B9', size = 2) + #add scatterplot points
  geom_smooth(method = lm, color='#2C3E50') + #confidence bands
  geom_line(aes(y = .lower, color = "CI"), linetype = "dashed") + #lwr conf interval
  geom_line(aes(y = .upper, color = "CI"), linetype = "dashed") + #upr conf interval
  geom_line(aes(y = PI.lwr, color = "PI"), linetype = "dashed") + #lwr pred interval
  geom_line(aes(y = PI.upr, color = "PI"), linetype = "dashed") + #upr pred interval
  labs(x = "Predictors", y = "Length of stay (LOS)", color = "",
       title = "Plots showing CI and PI bands on the models for each predictor") +
  scale_color_manual(values = colors) +
  facet_wrap(~predictor, scales = "free") +
  theme(legend.position = "bottom" )
```




### 0.3 2.65: Interval estiamtes of $\beta_1$ for each region

```
# Create a new dataset for convenient modeling.
senic_nested2 <- senic %>%
  # Select variables of interest
  select(LOS, Infection_Risk=infec_risk, region) %>%
  nest(-region)

# Fit a linear regression model between LOS and infection for each region.
```

```
models2 <- senic_nested2 %>%
  mutate(model = map(data, ~ lm(LOS ~ Infection_Risk, data = .)))

mdl_result <- models2 %>%
  mutate(metrics = map(model, tidy, conf.int = T, conf.level = 0.95)) %>%
  unnest(metrics) %>%
  filter(term == "Infection_Risk") %>%
  select(-c(data, model, statistic, p.value)) %>%
  rename(slope = estimate) %>%
  mutate(CI.length = conf.high - conf.low) %>%
  mutate(region_label = factor(region, levels = 1:4,
                       labels = c("NE", "NC", "S", "W")),
         .after = "region") %>%
  arrange(region)

# Display results
kable(mdl_result, escape = F, align = "c",
      caption = "Model results for each region with 95 % confidence intervals of $\\beta_1$")
```

Table 3: Model results for each region with 95 % confidence intervals of $\beta_1$

| region | region_label | term | slope | std.error | conf.low | conf.high | CI.length |
|---|---|---|---|---|---|---|---|
| 1 | NE | Infection_Risk | 1.3477 | 0.3159 | 0.6984 | 1.9970 | 1.2986 |
| 2 | NC | Infection_Risk | 0.4832 | 0.1366 | 0.2041 | 0.7622 | 0.5581 |
| 3 | S | Infection_Risk | 0.5251 | 0.1107 | 0.3003 | 0.7499 | 0.4496 |
| 4 | W | Infection_Risk | 0.0173 | 0.3058 | -0.6387 | 0.6732 | 1.3119 |

The information in the above table suggests that the regression lines for the different regions have different slopes. This is due to the differences in the magnitude of the slopes and largely due to the variability around each slope. Notwithstanding, we do see similarities between pairs of regions, for instance, the regression lines for regions **2** and **3** appear to have similar slopes, and so do regions **1** and **4**, as clearly indicated by the length of the confidence intervals.

### 0.3.1 CI and PI bands on the models

```
# Append confidence/prediction estimates for the slopes to the data for plotting.
PI_df_region <- models2 %>%
  mutate(newdata = map(model, augment, interval = "prediction")) %>%
  unnest(newdata) %>%
  select(PI.lwr = .lower, PI.upr = .upper)

senic_region_augmented <- models2 %>%
  mutate(newdata = map(model, augment, interval = "confidence")) %>%
  unnest(newdata) %>%
  bind_cols(PI_df_region) %>%
  mutate(region_label = factor(region, levels = 1:4,
                       labels = c("Region 1: NE", "Region 2: NC", "Region 3: S", "Region 1: W")),
         .after = "region")

#----create plot-----
# define colors to help add legend
colors <- c("CI"="green", "PI"="coral2")
ggplot(senic_region_augmented, aes(x = Infection_Risk, y = LOS)) +
```

```
  geom_point(color='#2980B9', size = 2) + #add scatterplot points
  geom_smooth(method = lm, color='#2C3E50') + #confidence bands
  geom_line(aes(y = .lower, color = "CI"), linetype = "dashed") + #lwr conf interval
  geom_line(aes(y = .upper, color = "CI"), linetype = "dashed") + #upr conf interval
  geom_line(aes(y = PI.lwr, color = "PI"), linetype = "dashed") + #lwr pred interval
  geom_line(aes(y = PI.upr, color = "PI"), linetype = "dashed") + #upr pred interval
  labs(x = "Infection Risk", y = "Length of stay (LOS)", color="",
       title = "Plots showing CI and PI bands on the models for each region") +
  scale_color_manual(values = colors) +
  facet_wrap(~region_label, scales = "free") +
  theme(legend.position = "bottom")
```