Bennett Hellman

15.072: Advanced Analytics Edge Fall 2021
Homework 4: From Predictions to Prescriptions

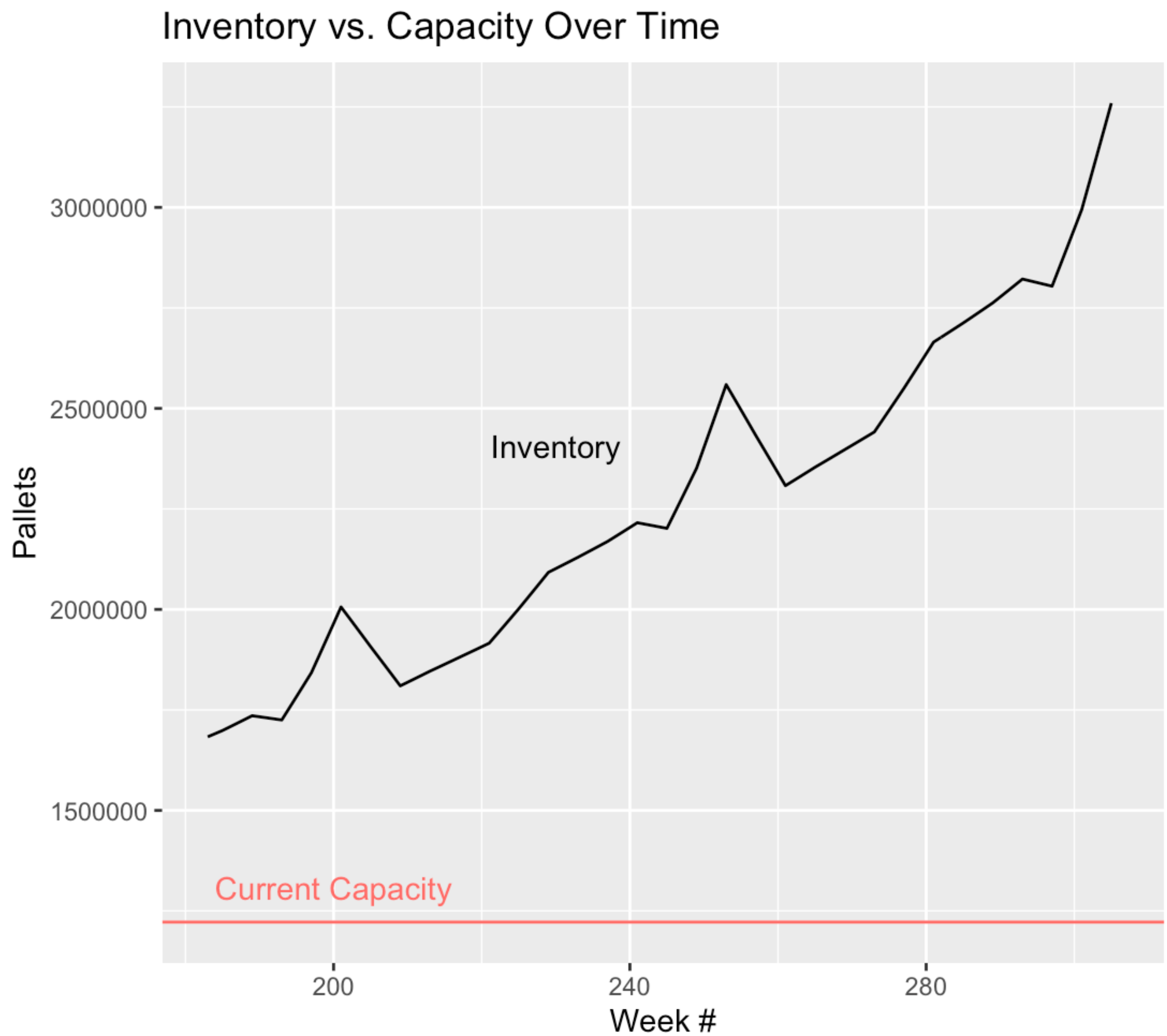
a.i)

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.312e+00  3.317e-02  69.712   <2e-16 ***
linc        -2.351e-01  3.147e-03 -74.711   <2e-16 ***
Week_Num     4.771e-03  2.832e-05 168.495   <2e-16 ***
bseason2     2.167e-03  3.757e-03   0.577    0.564    
bseason3     2.182e-03  3.762e-03   0.580    0.562    
bseason4     1.750e-03  3.770e-03   0.464    0.643    
bseason5     1.315e-03  3.782e-03   0.348    0.728    
bseason6     5.064e-02  3.797e-03  13.336   <2e-16 ***
bseason7     5.318e-02  3.816e-03  13.936   <2e-16 ***
bseason8     4.891e-02  3.838e-03  12.744   <2e-16 ***
bseason9     5.253e-02  3.863e-03  13.600   <2e-16 ***
bseason10    5.409e-02  3.891e-03  13.903   <2e-16 ***
bseason11    2.517e-03  3.922e-03   0.642    0.521    
bseason12    1.423e-01  3.956e-03  35.974   <2e-16 ***
bseason13    1.410e-01  3.993e-03  35.317   <2e-16 ***
```

```
his<-his%>%mutate(lsale = log(Sales/Population),saleper = Sales/Population,
              linc = log(Income), bseason = as.factor(Season))
train <- his%>%filter(Year <=2013)
test<-his%>%filter(Year ==2014)
mod <- lm(lsale~linc+Week_Num+bseason, data = train)
summary(mod)
```

a.ii)
It is important to note that none of these estimates are causal and can only be interpreted as correlation. A one percent increase in income per capita is associated with a 0.235% decrease in sales per capita. Another week away from the beginning of 2012 is associated with a .477% increase in sales per capita. Using season 2 as an example, being in season 2 is associated with a .217% increase in sales per capita. All other seasons can be interpreted the same was a 100*coefficient increase in sales per capita.

a.iii) **Out-of-sample performance: R^2 = 0.777, MAE = 0.163, RMSE = 0.209** b.i)



b.ii) **Shortage = 166.674%**

```
percent_cap_short <- (inv_df[123,3]-current_cap)/(current_cap)*100
```

c.i)
Decision Variables:
$b_i = 1\ if\ DC_i\ is\ chosen, 0\ otherwise$
$u_i = 1\ if\ DC_i\ \ is\ serves\ County_j, 0\ otherwise$
$c_i = the\ capacity\ utilized\ of\ DC_i$

Inputs:
$FixedCost_i = fixed\ cost\ of\ constructing\ DC_i$
$VariableCost_i = variable\ cost\ of\ constucting\ DC_i\ per\ sqft$
$d_{ij} = distance\ from\ DC_i\ to\ county_j$
$CapacityDC_i = maximum\ capacity\ of\ DC_i$
$x_{jt} = predicted\ demand\ of\ county\ j\ in\ week\ t$

$$\mathrm{Min}_{\mathrm{b,u,c}} \sum_{\mathrm{i}} FixedCost_i b_i + \sum_{\mathrm{i}} VariableCost_i c_i + \sum_{\mathrm{i}} \sum_{\mathrm{j}} \sum_{\mathrm{t}} \left(\frac{1.55}{20}\right) d_{ij} x_{jt} u_{ij}$$
$$b_i = 1\ \forall\ i = 1,2,3$$
$$c_i = 1{,}200{,}000\ \forall\ i = 1,2$$
$$c_i = 900{,}000\ \forall\ i = 3$$
$$\sum_{j}^{20} u_{ij} = 1\ \forall\ j$$
$$u_{ij} \leq b_i\ \forall\ i, j$$
$$c_i = 1{,}200{,}000\ \forall\ i = 1,2$$
$$0 \leq c_i \leq CapacityDC_i b_i$$
$$\sum_{j} u_{ij} \leq CapacityDC_i \left(\frac{5}{13.5}\right) \forall\ i$$
$$c_i \leq 1{,}200{,}000 b_i \forall\ i$$
$$u_{ij}, b_i \in \{0,1\}$$

```
model = Model(with_optimizer(Gurobi.Optimizer, Gurobi.Env()))

set_optimizer_attribute(model, "OutputFlag", 0)

n = size(county_tot_d,1)

@variable(model, b[i=1:20], Bin)
@variable(model, u[i=1:20,j=1:n], Bin)
@variable(model, c[i=1:20]>=0)


@constraint(model, [i=1:3], b[i]==1)
@constraint(model, [i=1:2], c[i]==1200000)
@constraint(model, [i=3], c[i]==900000)
@constraint(model, [j=1:n], sum(u[:,j])== 1)
```

```
@constraint(model, [i=1:20, j=1:n], u[i,j] <= b[i])
@constraint(model, [i=1:20], c[i]*(5/13.5) >= sum(df[j]*u[i,j] for j=1:n))
@constraint(model, [i=1:20], c[i]<=b[i]*1200000)

@objective(model, Min, sum(variable_cost[i]*c[i] for i=1:20) + sum(fixed_cost[i]*b[i]
for i=1:20) +
   sum(sum((1.55/20)*d_mat[i,j]*county_tot_d[j,:d_pallets_sum]*u[i,j] for j=1:n) for
i=1:20))

optimize!(model)
```

d.i) **In addition to the 3 original DCs, there will also be DCs at Kalamazo, Lancaster, Scranton, Syracuse, Toledo.**

```
b=value.(b)
```

d.ii)
Providence = 57
Richmond = 122
Youngstown = 120
Kalamazo = 222
Lancaster = 41
Scranton = 27
Syracuse = 60
Toledo = 116

```
sum(u, dims=2)
```

d.iii)
Construction Cost = $556,050,072.64
Transportation Cost = $197,324,788.82

```
construction_cost = sum(variable_cost[i]*c[i] for i=1:20) + sum(fixed_cost[i]*b[i] for
i=1:20)
transpo_cost = sum(sum((1.55/20)*d_mat[i,j]*county_tot_d[j,:d_pallets_sum]*u[i,j] for
j=1:n) for i=1:20)
```

e.i)
Cheapest DC’s w/ capacity (in sqft)
Providence = 1,200,000
Richmond = 1,200,000
Youngstown = 900,000
Bangor = 1,200,000
Burlington = 1,200,000
Dover = 1,200,000

Kalamazoo = 700237.0249
Worcester = 1,200,000

Construction Costs = $517,077,966.49
Transportation Costs = $505,919,172.89

```
mod1 = Model(with_optimizer(Gurobi.Optimizer, Gurobi.Env()))
set_optimizer_attribute(mod1, "OutputFlag", 0)

n = size(county_tot_d,1)

@variable(mod1, c[i=1:20]>=0)
@variable(mod1, b[i=1:20], Bin)

@constraint(mod1, [i=1:2], c[i]==1200000)
@constraint(mod1, [i=3], c[i]==900000)
@constraint(mod1, [i=1:3], b[i]==1)
@constraint(mod1, [i=1:20], c[i]<=b[i]*1200000)
@constraint(mod1, sum(c[i]*(5/13.5) for i=1:20) >= sum(df[j] for j=1:n))

@objective(mod1, Min, sum(fixed_cost[i]*b[i] for i=1:20) + sum(variable_cost[i]*c[i]
for i=1:20))

optimize!(mod1)

mod2 = Model(with_optimizer(Gurobi.Optimizer, Gurobi.Env()))
set_optimizer_attribute(mod2, "OutputFlag", 0)

n = size(county_tot_d,1)

@variable(mod2,u[i=1:20,j=1:n], Bin)

b = value.(b)
c = value.(c)

@constraint(mod2, [j=1:n], sum(u[:,j])== 1)
@constraint(mod2, [i=1:20, j=1:n], sum(df[j]*u[i,j] for j=1:n) <= 1.001*c[i]*5/13.5)

@objective(mod2, Min,
sum(sum(1.55/20*d_mat[i,j]*county_tot_d[j,:d_pallets_sum]*u[i,j] for j=1:n) for
i=1:20))

optimize!(mod2)
```

e.ii) **In the baseline solution, which first optimizes for construction cost and then generates transportation costs, the solution reflects the priorities. Construction cost is lower in the**

second model because it is explicitly minimizes it first. However, this comes at the cost of higher transportation costs. In part d, we were only concerned with the overall cost and thus that was decreased, and transportation costs were significantly less.

Bennett Hellman

R-Code Appendix

```
library(tidyverse)
library(Metrics)
library(zoo)

his = read.csv("/Users/bennetthellman/Desktop/OneDrive - Massachusetts Institute of
Technology/AE/HWs/HW4/Dartboard_historical-1.csv");
fut = read.csv("/Users/bennetthellman/Desktop/OneDrive - Massachusetts Institute of
Technology/AE/HWs/HW4/Dartboard_future.csv.crdownload");
dc = read.csv("/Users/bennetthellman/Desktop/OneDrive - Massachusetts Institute of
Technology/AE/HWs/HW4/Dartboard_dcs.csv");

#dc<-dc%>%mutate(County.Name = Location)

#mdf = merge(his, dc, by.x ="State.Name", by.y = "Location", all.x = TRUE)

his<-his%>%mutate(lsale = log(Sales/Population),saleper = Sales/Population,
          linc = log(Income), bseason = as.factor(Season))
train <- his%>%filter(Year <=2013)
test<-his%>%filter(Year ==2014)

mod <- lm(lsale~linc+Week_Num+bseason, data = train)
summary(mod)
summary(mod)$r.squared

pred <- predict(mod, newdata=test)
#OSR^2
1 - sum((pred - test$lsale)^2) / sum((mean(train$lsale) - test$lsale)^2)
#OMAE
mae(test$lsale, pred)
#ORMSE
rmse(test$lsale, pred)

fut<-fut%>%mutate(linc = log(Income), bseason = as.factor(Season))
fut$forecast <- exp(predict(mod, newdata=fut))*fut$Population

#b
inv_df <- fut %>% group_by(Week_Num) %>% summarize(tot_d= sum(forecast)) %>%
  mutate(inv = (rollapply(data = tot_d, FUN=sum, width=8, align="left", fill=NA)/1000))
#Claire Sailard helped me with this function
current_cap<- dc%>%summarise(tot_pallets_cap = sum(5*Current_Size/(13.5)))

ggplot(main = "Inventory vs. Capacity Over Time")+geom_line(aes(x = Week_Num, y = inv),
data = inv_df) + geom_hline(aes(yintercept=1222222, color = "red")) +
  geom_text(aes(200,1222222,label = "Current Capacity", vjust = -1, color = "red")) +
geom_text(aes(230,2322222,label = "Inventory", vjust = -1))+
```

```
 labs(x ="Week #", y="Pallets", title="Inventory vs. Capacity Over Time")+
theme(legend.position = "none")+xlim(183,306)

#bi
percent_cap_short <- (inv_df[123,3]-current_cap)/(current_cap)*100
percent_cap_short

#exportation
fut$d_pallets = fut$forecast/1000
write.csv(fut,'/Users/bennetthellman/Desktop/OneDrive - Massachusetts Institute of
Technology/AE/HWs/HW4/pred_d.csv')
```

Bennett Hellman

Julia Code Appendix