Assignment 1

Group 11

Chang Zhou 01983512, Qian Zhang 01939418, Yutong Zheng 01895402

2021/5/10

Question 1

(a) $\phi(x) = x_1^2 + x_2^2$

(b) $\phi(x) = x_1 x_2$

(c) $\phi_1(x) = x_1$, $\phi_2(x) = x_1^2$

Question 2

(a)

```
data <-read.csv(file = "Tahoe_Healthcare_Data.csv", header = TRUE)
allad<-sum(data$readmit30) # 998 (1: readmitted, 0: not readmitted)
none<-allad*8000
none
```

```
## [1] 7984000
```

Taking the data-set as representative of what will happen in a given year if nothing is done to reduce the readmissions rate, there are 998 re-admitted patients. Given that the estimated loss in Medicare reimbursements would rise to $8,000 per re-admitted patient, we obtain the total cost as $7,984,000.

(b)

```
count<-count(data)
# Total cost if CareTracker was implemented for all AMI patients
max<-count*1200+allad*8000*.6  # CareTracker cost $1,200 per patient,
                               # can reduce the incidence of readmissions by 40%
netprofit<-none-max # Net change in cost
netprofit
```

```
##           n
## 1 -2064800
```

Tahoe should not implement CareTracker for all AMI patients because the net profit is -2064800, which is much smaller than 0.

(c)

```r
# Cost when nothing done - Cost when implementing CareTracker
# for exactly 998 re-admitted patients
none-sum(data$readmit30)*(8000*.6+1200)
```

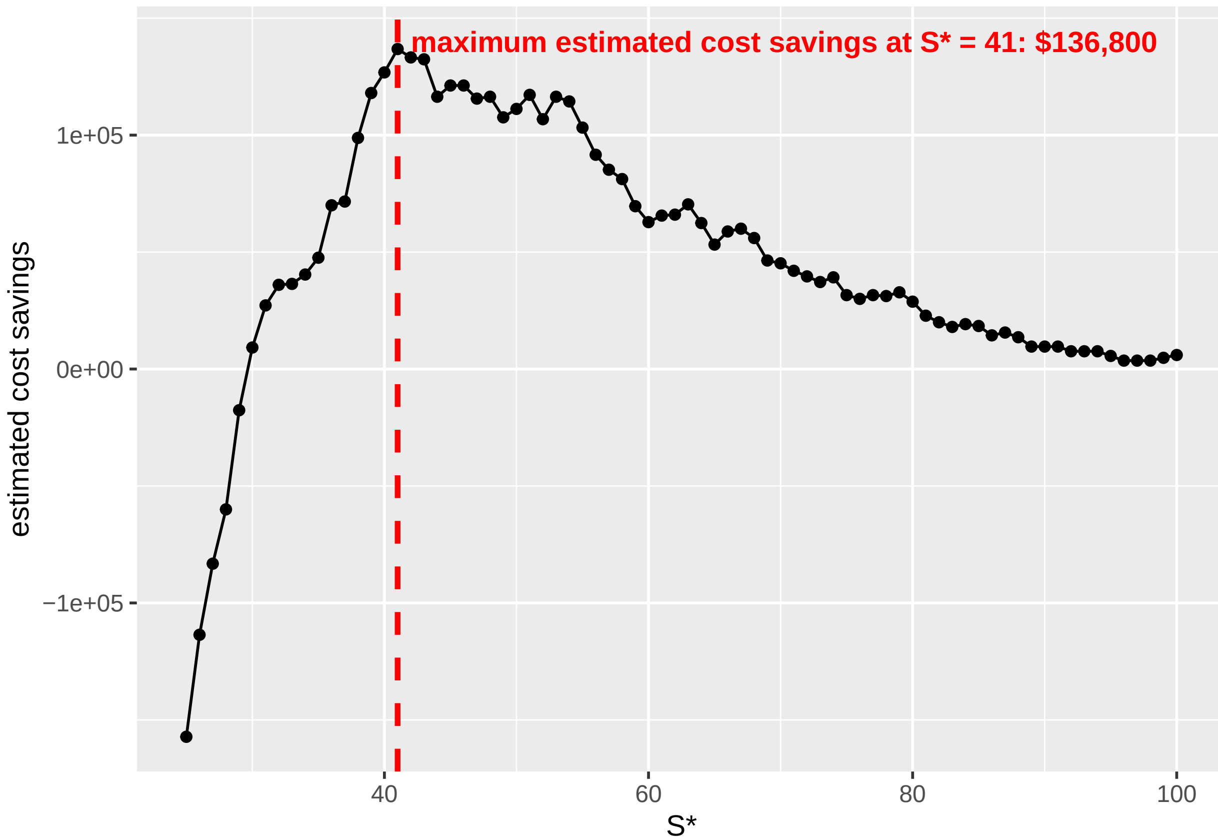
```
## [1] 1996000
```

If Tahoe had perfect foresight regarding re-admitted patients, they could explicitly implement CareTracker for 998 re-admitted patients and reduced the incidence of readmissions among these patients by 40%. Therefore, they will save $1,996,000 as a upper bound.

(d)

```r
x<- c(25:100)
list1 = c() # store cost savings
for (i in 25:100) {
  # filter out the patients with higher severity score
  T1<-data[data$severity.score>i,]
  # number of true re-admitted patients among the selected patients
  ad<-sum(T1$readmit30)
  # saved cost - implementation cost = final cost saving
  list1 <- c(list1, ad*8000*.4-dim(T1)[1]*1200)
}

ggplot(mapping = aes(x = seq(25, 100), y = list1)) +
  geom_point() + scale_x_continuous(limits=c(25, 100)) +
  geom_line() + labs(x = 'S*' , y = 'estimated cost savings') +
  geom_vline(xintercept=41, color = "red", size = 1,
             linetype="dashed") +
  annotate("text", x = 42, y = 140000, hjust = 0, fontface = 2,
           label= paste("maximum estimated cost savings at S* = 41: $136,800"), color = "red")
```



The best value for the threshold S^* is 41 with approximate cost saving $136,800.

(e)

```
glm.fit = glm(readmit30~.,data=data,family=binomial(link="logit"))
# summary(glm.fit)
stargazer(glm.fit, header = FALSE, type = 'latex', title = "2 (e)")
```

Let π denote the probability that $readmit30 = 1$, then from the model we constructed, we have

$$logit(\hat{\pi}) = -4.016 + 0.002 * age + 0.190 * female + 0.743 * flu\_season - 0.159 * ed\_admit$$

$$+0.027 * severity.score + 0.016 * comorbidity.score$$

(f)

```
# patient's estimated probability of readmission
glm.probs = predict(glm.fit,type="response")
data$p<- glm.probs
y<- seq(0.1, 0.9, 0.01)
list2 <- c() # store cost savings
for (i in 10:90) {
  # filter out the patients with higher estimated prob. being re-admitted
  T1<-data[data$p>i/100,]
  # number of true re-admitted patients among the selected patients
  ad<-sum(T1$readmit30)
  # saved cost - implementation cost = final cost saving
  list2 <- c(list2, ad*8000*.4-dim(T1)[1]*1200)
}
Y<- data.frame(seq(0.1, 0.9, 0.01), list2)
```

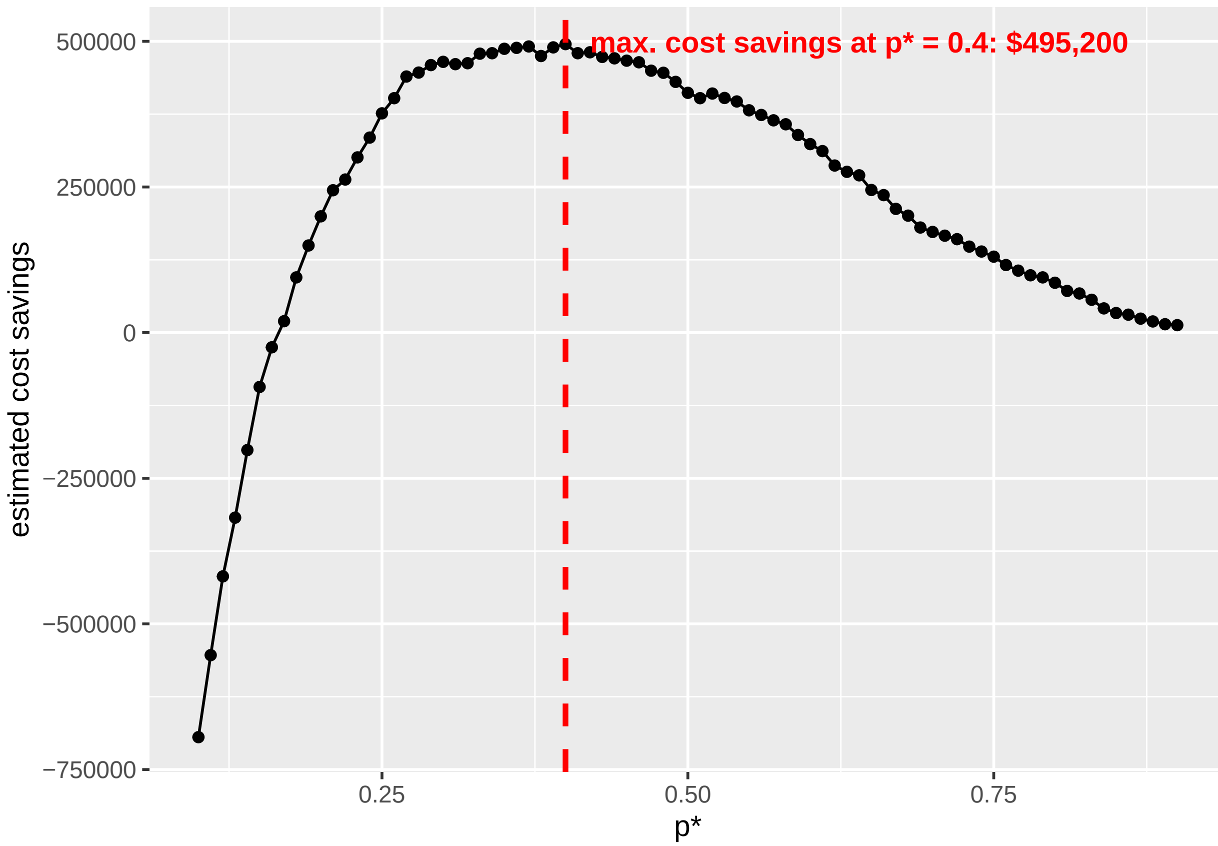
Table 1: 2 (e)

	Dependent variable:
	readmit30
age	0.002
	(0.005)
female	0.190^{**}
	(0.082)
flu_season	0.743^{***}
	(0.082)
ed_admit	-0.159
	(0.115)
severity.score	0.027^{***}
	(0.002)
comorbidity.score	0.016^{***}
	(0.001)
Constant	-4.016^{***}
	(0.410)
Observations	4,382
Log Likelihood	$-1,915.831$
Akaike Inf. Crit.	3,845.662
Note:	*p<0.1; **p<0.05; ***p<0.01

```r
colnames(Y)[1] = 'p'
colnames(Y)[2] = 'saving'
# Find out the maximum cost saving
Y[which.max(Y$saving), ]
```

```
##      p saving
## 31 0.4 495200
```

```r
ggplot(mapping = aes(x = seq(0.1, 0.9, 0.01), y = list2)) +
  geom_point() + scale_x_continuous(limits=c(0.1, 0.9)) +
  geom_line() + labs(x = 'p*' , y = 'estimated cost savings') +
  geom_vline(xintercept = 0.4, color = "red", size = 1,
             linetype="dashed") +
  annotate("text", x = 0.42, y = 499000, hjust = 0, fontface = 2,
           label= paste("max. cost savings at p* = 0.4: $495,200"), color = "red")
```



The best value for the threshold p^* is 0.4 with approximate cost saving $495,200.

Question3

(i) True

Logistic regression is one of the discriminative classification algorithms, and it fits a model of the form P(Y|**X**), which is the probability that Y belongs to a particular category (0 or 1) given X.

(ii) True

The goal of logistic regression is to estimate the conditional probability $P(Y = 1|X = x)$. Writting $P(Y = 1|X = x)$ as p, the likelihood is $\Pi_{i=1}^{N} pi(w)^{y_i}(1 - p_i(w))^{1-y_i}$. We will obtain the value of w by maximizing the log-likelihood.

(iii) False

Logistic regression is one of the discriminative classification algorithms, and it fits a model of the form P(Y|**X**) not P(**X**,Y).

(iv) True

The logistic Regression defines $P(Y = 1|X = x)$ to be $\frac{exp(w^T x)}{1+exp(w^T x)}$. $Y = 1$ if $P(Y = 1|X = x) > 0.5$, and $Y = 0$ if $P(Y = 1|X = x) < 0.5$. The function could be rewritten as $\frac{e^{w^T x}}{1+e^{w^T x}} = \frac{\frac{e^{w^T x}}{e^{w^T x}}}{\frac{1+e^{w^T x}}{e^{w^T x}}} = \frac{1}{e^{-w^T x}+1}$. From the final expression, it's clear that if $-w^T x$ is positive, the expression $\frac{1}{e^{-w^T x}+1}$ is less than 0.5 and Y is estimated to be 0. Also, if $-w^T x$ is negative, the expression $\frac{1}{e^{-w^T x}+1}$ is greater than 0.5 and Y is estimated to be 1. Therefore, the decision boundary is $-w^T x$. As it is a linear expression, logistic regression always produces a linear classifier.