Laboration 3: Statistisk hypotesprövning

Statistisk analys

Emil Erikson och Leo Levenius*

2024-06-12

Viktigt: Innan ni läser vidare

Gör först följande:

1. (Om ni inte gjort det tidigare) I den övre menyn i `RStudio`, tryck på `Tools --> Global Options --> Code --> Saving`. Under "Default text encoding", tryck på "Change" och välj "UTF-8".
2. Gå in på kurshemsidan och ladda ner mallen för denna laboration. Döp den till `labb3-efternamn1-efternamn2.Rmd`. Öppna den i Rstudio. Skriv er rapport i denna fil.

Krav för laboration 3

Tänk på följande **krav** på er rapport:

- Rapporten måste kunna läsas av någon som inte har läst labbinstruktionerna. Så ni måste skriva vad det är ni ska göra innan ni gör det, och berätta vad syftet är.
- Rapporten måste vara skriven i `R Markdown`.
- All kod som används måste synas i labbrapporten, men ska inte beskrivas i detalj i rapporten.
- Alla **tabeller och diagram** måste förses med **numrering och beskrivande text**, och refereras till i rapportens vanliga text på rätt sätt. Diagram måste ha lämpliga rubriker på axlarna och tabeller lämpliga rubriker på kolumner. *Tips*: Använd `fig.cap` för automatisk numrering.

*Tidigare versioner av Jakob Torgander, Benjamin Allévius och Fredrik Olsson.

Sammanfattning av laboration 3

I sista datorövningen kommer ni att använda enkel linjär regression för att studera egenskaperna hos ett material bestående av bivariata data. Precis som i tidigare övningar kommer först en inledande beskrivning av hur dessa metoder ser ut i R, och sedan följer själva uppgifterna.

Enkel linjär regression och korrelationsanalys i R

Enkel linjär regression

Enkel linjär regression hanteras i `R` av funktionen `lm`. Skriv `?lm` i Console för en detaljerad beskrivning av hur denna funktion är uppbyggd. Om ni vill läsa beskrivningen i er webbläsare istället för i RStudio kan ni först skriva `help.start` i Console, så kommer sidan för `?lm` visas i webbläsaren.

Som ni kan se är detta en väldigt kraftfull funktion som kan hantera betydligt mer komplicerade modeller än enkel linjär regression, men i den här datorövningen kommer vi endast att använda den i det specialfallet.

Låt oss anta att variablerna vi ska använda oss av i vår regression finns inlästa i `R` som en `data.frame`, vilket är en sorts tabell där varje kolonn tillåts ha en egen datatyp. Med datatyp menas här t.ex. `numeric` (decimaltal), `integer` (heltal), `character` (textsträngar). Vanligtvis representera kolonnerna olika variabler, och raderna olika observationer.

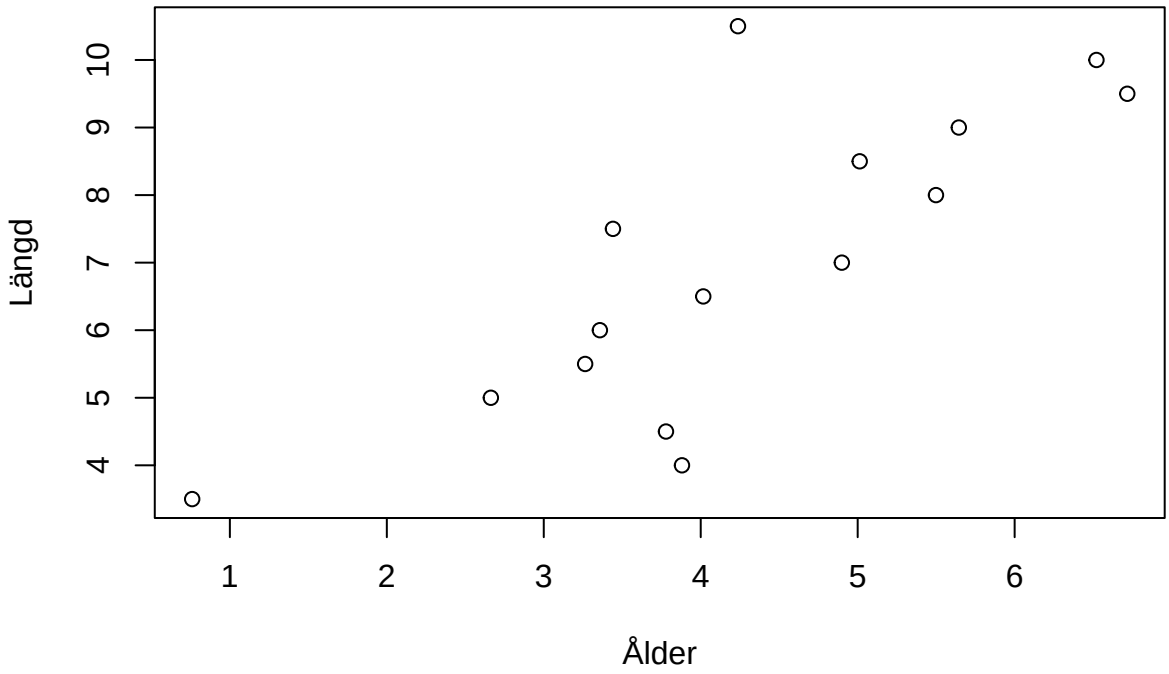
Som ett exempel, låt Y beteckna längden i meter av en visst slags träd som växer i en skogsdunge. Antag att längden av ett träd beror linjärt på dess ålder x (år), men att längden också påverkas (additivt) av en rad andra mindre viktiga faktorer som vi buntar ihop och låter representeras av ett standardnormalfördelat brus ϵ. Vår modell är alltså $Y = \alpha + \beta x + \epsilon$. Antag sedan att vi har ett stickprov (x_i, y_i), $i = 1, \ldots, 15$, med vilket vi vill skatta parametrarna α och β. Stickprovet har lagts i en `data.frame`, vilket är en sorts tabell i `R` som är smidig att jobba med när man anpassar modeller och liknande. I detta exempel har data framen döpts till `df`, med kolonner `length` och `age`, som motsvarar Y respektive x i modellen ovan. Den ser ut på följande vis:

```
df
```

```
      length  age
1  0.7598822  3.5
2  3.8805845  4.0
3  3.7786069  4.5
4  2.6627866  5.0
5  3.2636349  5.5
6  3.3574873  6.0
7  4.0161055  6.5
8  4.8990113  7.0
9  3.4408208  7.5
10 5.4987910  8.0
11 5.0128385  8.5
12 5.6444853  9.0
13 6.7177647  9.5
14 6.5212941 10.0
15 4.2372980 10.5
```

Datan ser alltså ut på följande sätt om vi plottar den:

```
plot(df$length, df$age, xlab = "Ålder", ylab = "Längd")
```



Figur 1: Längd plottad mot ålder för träden i skogsdungen.

Här är Y alltså vår **responsvariabel**[1] och x är den **förklarande variabeln**[2]. Vi ska nu göra en linjär regression av Y på x. I `R` kan vi göra detta enkelt med kommandot

```
modell <- lm(length ~ 1 + age, data = df)  # anpassa modellen
```

Här säger vi att datan som ska användas är den som finns i data framen `df`, och vi kan därför referera till namnen på kolumnerna i denna data frame i formeln `length ~ 1 + age`. Här har vi med en etta för att göra det tydligt att vi vill ha ett intercept (motsvarande α) i vår modell. Byter man ut ettan mot en nolla så blir det en modell utan intercept.

VIKTIGT: notera att namnen `length` och `age` är dem jag valde när jag skapade data framen `df`. De är specifika för just detta exempel, och jag kunde ha valt andra namn om jag så velat. Så ni kan inte kopiera denna kod till en uppgift rakt av och tro att det kommer att fungera.

[1] Beroende på sammanhang, även kallad beroende variabel, förklarad variabel, med mera.
[2] Beroende på sammanhang, även kallad oberoende variabel, bakgrundsvariabel, prediktorvariabel, med mera.

Vi kan inspektera den anpassade modellen med hjälp av kommandot `summary`:

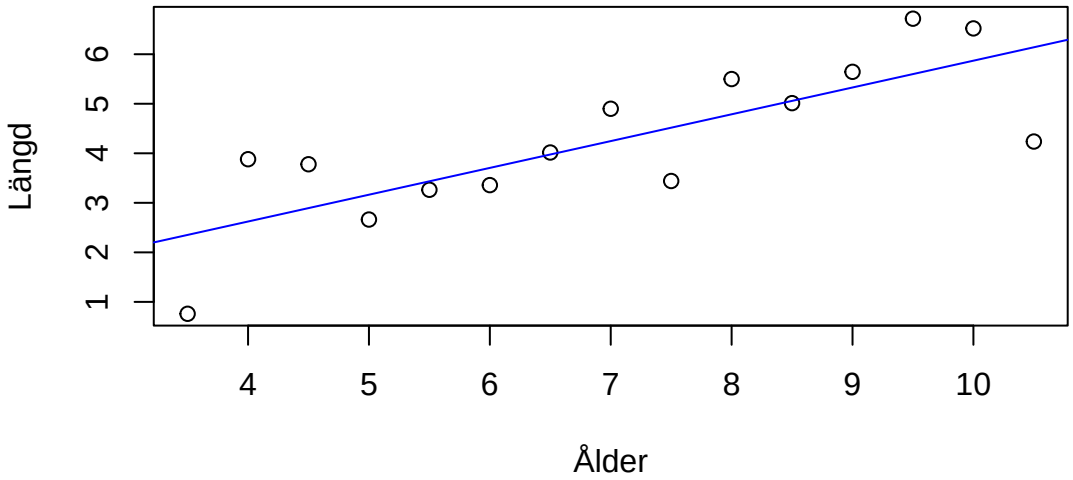
```
summary(modell)  # visa resultatet
```

```
Call:
lm(formula = length ~ 1 + age, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-1.9029 -0.4242  0.0406  0.6822  1.2580

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.4579     0.8686   0.527 0.606948
age           0.5412     0.1186   4.564 0.000531 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.992 on 13 degrees of freedom
Multiple R-squared:  0.6157,    Adjusted R-squared:  0.5862
F-statistic: 20.83 on 1 and 13 DF,  p-value: 0.0005312
```

Där det står `Coefficients` så ges i kolumnen `Estimate` skattningar av parametrarna α (intercept) och β (lutningskoefficient), och i kolumnen `Pr(>|t|)` så ges p-värden för tvåsidiga test av hypoteserna $H_0 \colon \alpha = 0$ och $H_0 \colon \beta = 0$. Tyvärr finns inget enkelt sätt att få `R` att automatiskt genomföra ensidiga test, men man kan enkelt få p-värden för ett ensidigt test genom att halvera p-värdet för motsvarande tvåsidiga test. Man får också skattningen s^2 av variansen som `Residual standard error` och förklaringsgraden R^2 som `Multiple R-Squared`.

Vi kan rita in den anpassade linjen i plotten ovan på följande sätt:

```
plot(df$age, df$length, xlab = "Ålder", ylab = "Längd")
abline(modell, col = "blue")
```



Figur 2: Anpassad modell.

Residualanalys

Modellens residualer är observationernas (vertikala) avvikelse kring linjen ovan. I och med att residualerna kan visas skatta feltermerna $\varepsilon_i = Y_i - (\alpha + \beta x_i)$, så kan vi genom att inspektera residualerna undersöka om förutsättningarna för modellen är uppfyllda. För att få residualerna från en anpassad modell kan man enkelt skriva

```
residual <- modell$residuals
```

En residualplot fås nu med

```
plot(df$age, residual)
abline(a = 0, b = 0, lty = "dotted")
```

Det andra kommandot ovan ritar in en prickad ("dotted") linje längs nollan, vilket kan underlätta avläsning av en residualplot. Slutligen fås en normalfördelningsplot av residualerna på vanligt sätt enligt

```
qqnorm(residual)
qqline(residual)
```

Prediktion

Linjära regressionsmodeller kan även användas för prediktion av ny data, dvs att "förutspå" nya värden. Prediktion av ny data är bland annat det centrala momentet inom så kallad övervakad maskininlärning (supervised machine learning), där linjär regression utgör en viktig basmodell. Prediktion kan i `R` utföras med hjälp av funktionen `predict`. Denna funktion tar in en anpassad modell, tillsammans med den nya datan som vi vill prediktera för.

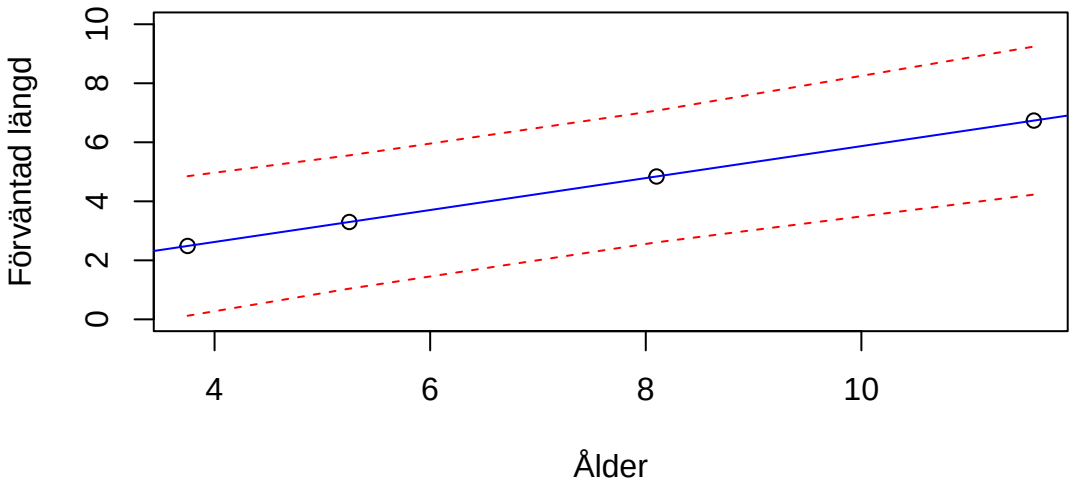
Antag till exempel att vi efter att ha anpassat vår trädlängdsmodell får information om ålder för fyra nya träd. Den predikterade längden för dessa nya träd kan då beräknas med hjälp av följande kodstycke:

```
x_new <-  c(3.75, 5.25, 8.1, 11.6) # Nya trädåldrar
df_new <- data.frame(age = x_new)
predictions <- predict(modell, newdata= df_new, interval = 'predict')
```

Observera att variabelnamnen för den nya datan måste överensstämma med namnen från den ursprungliga datan (i detta fall "age") för att `predict` ska fungera. Genom att lägga till argumentet interval = 'predict' så returnerar `predict` både predikterade värden samt övre resp. undre gränser för ett 95% prediktionsintervall (om inget argument sätts returnerar funktionen endast predikterade värden).

Vi kan nu jämföra våra predikterade värden med regressionslinjen från vår modell enligt följande kodstycke:

```
# Plockar ut prediktioner samt gränser på prediktionsintervallet
y_pred <- predictions[, 1]
interval_lower <- predictions[, 2]
interval_upper <- predictions[, 3]

# Plottar predikterade värden tillsammans med prediktionsintervall
# samt regressionslinje
plot(x_new, y_pred, xlab = "Ålder", ylab = "Förväntad längd",  ylim = c(0,10))
lines(x_new,interval_lower, col = "red", lty = "dashed")
lines(x_new, interval_upper, col = "red", lty = "dashed")
abline(modell, col = "blue")
```



Figur 3: Predikterade värden från regressionsmodellen

Vi ser i denna figur att de predikterade värdena mycket riktigt ligger längs regressionslinjen till vår anpassade modell. I figuren är även vårt 95% prediktionsintervall illustrerat med röda streckade linjer.

Modellutvärdering

Ett vanligt mått för att utvärdera hur väl en modell predikterar givet ny, tidigare osedd data är att genom att beräkna det så kallade medelkvadratfelet (Mean-Squared Error) mellan predikterade värden $\hat{y}_i$ och faktiska värden y_i. medelkvadratfelet ges av följande uttryck

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2,$$

där ett högt fel indikerar att de predikterade värdena i snitt ligger långt ifrån de faktiska värden och vice versa. Medelkvadratfelet kan beräknas i `R` genom följande funktion:

```
mse <- function(y_actual, y_pred){
  diff <- y_actual - y_pred
  sqr_diff <- diff^2
  mse <- mean(sqr_diff)
  return(mse)
}
```

Om vi nu antar att vi får information om de *faktiska* längderna till våra fyra nya träd så kan vi därmed utvärdera prediktionerna från vår regressionsmodell enligt följande kodstycke.

```
y_actual <- c(1.39, 5.2, 4.09, 5.88) # Faktiska trädlängder
model_mse <- mse(y_actual, y_pred) # Beräknar MSE
print(paste("MSE:", round(model_mse,3)))
```

```
## [1] "MSE: 1.529"
```

För att få ett mått med samma enhet (i detta fall meter) kan vi därefter beräkna kvadratroten ur medelkvadratfelet. Detta ger oss det så kallade rot-medelkvadratfelet (Root Mean-Squared Error)

$$\text{RMSE} = \sqrt{\text{MSE}},$$

vilket för vår data kan beräknas enkelt genom

```
model_rmse = sqrt(model_mse)
print(paste("MSE:", round(model_rmse,3)))
```

```
## [1] "MSE: 1.236"
```

Vi får alltså ett genomsnittligt fel på 1.236 meter när vår linjära regressionsmodell applicerats på den nya datan.

Data frames och att "subsetta" (ta ut delmängder)

Ibland, som i Uppgift 2 nedan, så vill man arbeta med delmängder av sitt datamaterial. Nedan visas ett sätt att göra detta på, som använder sig av data framen `df` som vi definierade ovan. I detta exempel vill vi utföra linjär regression på datan i de tre delmängderna.

Säg att vi vill dela in data efter variabeln x, som representeras av kolumnen `age` i den data frame datan lagrats i. Vi vill ha tre perioder: $x \leq 5$, $5 < x \leq 10$, och $x > 10$. Om vi vill kan vi skapa tre nya data frames indelade efter dessa perioder:

```
df1 <- subset(df, age <= 5)
df2 <- subset(df, age > 5 & age <= 10)
df3 <- subset(df, age > 10)
```

Här säger vi t.ex. att `df1` ska bli en data frame bestående av de rader i data framen `df` för vilka kolumnen `age` är mindre än eller lika med 5. Vi använder oss av funktionen `subset`, som ju betyder delmängd på engelska.

Regressionen kan sedan utföras precis som ovan:

```
modell1 <- lm(length ~ age, data = df1)
modell2 <- lm(length ~ age, data = df2)
modell3 <- lm(length ~ age, data = df3)
```

Uppgift 1: Jordens medeltemperatur 1850–2007

I filen `temperatur.csv` på kursens hemsida finns data över avvikelsen av jordens medeltemperatur under perioden 1850–2007 jämfört med genomsnittet för perioden 1961–1990.[3] Börja med att ladda ner filen till mappen ni ska spara labben i (förslagsvis 'Documents/Kurser/statan/Labb3 eller liknande) och läs in den i R som förut med kommandot

```
df <- read.csv("temperatur.csv", header = TRUE)
```

Filen innehåller två kolumner: `år` som anger året, `temperatur` som anger den globala medeltemperaturen, Data kommer från Carbon Dioxide Information Analysis Center (http://cdiac.ornl.gov/).

Besvara följande frågor:

1. Gör en scatterplot med den globala medeltemperaturen (`temperatur`) på y-axeln och årtal (`år`) på x-axeln och kommentera sambandet. Ser det linjärt ut under hela perioden?
2. Ange förutsättningarna för enkel linjär regression.
3. Genomför en enkel linjär regression av den globala medeltemperaturen (`temperatur`) som responsvariabel och årtal (`år`) som förklarande variabel.
   - Gör samma scatterplot som ovan (igen), och rita in den anpassade linjen i plotten. Ser linjen ut att passa bra?
4. Anser ni att förutsättningarna för en enkel linjär regression är uppfyllda för datamaterialet? Motivera ert svar!

[3] Perioden 1961–1990 används internationellt som referensperiod när man ska avgöra om klimatet avviker på något sätt. När exempelvis TV-meteorologerna säger att medeltemperaturen under den kommande femdygnsperioden ligger över eller under det normala är det motsvarande period under 1961–1990 de jämför med.

- Visa och dra slutsatser från en residualplot och en normalfördelningsplot. Vilka förutsättningar från fråga 2 ovan stämmer?

Kom ihåg att alla **tabeller och diagram** måste förses med **numrering och beskrivande text**. *Tips*: Använd `fig.cap` för automatisk numrering.

Uppgift 2: Jordens medeltemperatur under tre perioder

Ett sätt att få modellen att passa bättre är att dela in hela mätperioden i ett antal delperioder. Vi ska nu dela in datamaterialet i tre separata tidsperioder, nämligen 1880–1929, 1930–1969, och 1970–2007.

Ni ska sedan göra följande:

1. Utför samma enkla linjära regression som sist med temperaturdatan, fast tre gånger om: en för varje indelning av variabeln `år` enligt de tidsperioder som angavs ovan. 1.1 För var och en av dessa tidsperioder för sig, undersök om förutsättningarna (för enkel linjär regression) nu är bättre uppfyllda. 1.2 För var och en av dessa tidsperioder för sig, ange punktskattningarna av α (intercept) och β (lutningskoefficient) för den enkla linjära regressionen.
2. En direktör för oljebolaget Fossil Fools hävdar: "Växthuseffekten är inget problem, det finns ingen bevisad trend mot ett varmare klimat". Testa denna hypotes på signifikansnivån 5% för perioden 1970–2007 genom att besvara följande frågor: 2.1 Bör testet vara ensidigt eller tvåsidigt? 2.2 Vad är kopplingen mellan direktörens påstående och modellen som anpassades för samma tidsperiod? (Bör leda er till vilket test som är lämpligt) 2.3 Vilka är nollhypotesen samt den alternativa hypotesen? Uttryck dessa i både ord och symboler. 2.4 Vilket sorts test är det som används? 2.5 Vad ger testet för svar? Förkastar vi nollhypotesen? Har direktören rätt eller fel?

Uppgift 3: Prediktion

I denna avslutande del så skall vi undersöka lämpligheten att använda vår anpassade regressionsmodell för att prediktera framtida temperaturer. Vi kommer här utvärdera vår modell på tidsperioden 2008-2022 vilka finns samlade csv-filen "temperatur_test", som läses in som förut enligt följande kodblock:

```
df_test <- read.csv("temperatur_test.csv", header = TRUE)
```

Denna fil innehåller samma två kolumner som innan.

Uppgift 3.1

Ni ska nu generera prediktioner för åren 2008–2022. Detta ska ni göra genom att ställa upp *två* linjära regressionsmodeller. Den ena ska använda hela perioden (1850–2007), och den andra ska använda åren 1970–2007. För båda modellerna, generera prediktioner för åren 2008–2022 och beräkna sedan *roten ur medelkvadratfelet* (RMSE) mellan de predikterade värdena och de faktiska temperaturvärdena i testdatan för dessa två prediktioner.

Besvara följande frågor:

1. Tolka resultatet. Är felet högt eller lågt? Motivera!
2. Vad blir skillnaden av att utvärdera felen på de olika datamängderna? Vad beror denna skillnad på?
3. Vad är fördelarna och nackdelarna med att inkludera mer data i modellanpassningen?

Upprepa nu detta, fast generera istället prediktioner för åren 1970–2007 (d.v.s. för *samma* data som ena modellen anpassades för). Besvara på de 3 ovan ställda frågorna **igen** för de två *nya* prediktionerna.

Samla sedan de fyra RMSE-värdena i en tabell.

Uppgift 3.2

Använd den linjära regressionsmodellen som var anpassad till åren 1970–2007. Plotta de predikterade värdena samt prediktionsintervallet för åren 2008-2022, tillsammans med de faktiska värdena för testdatan i ett spridningsdiagram. *Tips*: använd R-funktionen `points` för att lägga till punkter i en befintlig plott.

Besvara följande frågor:

1. Är linjär regression en lämplig prediktionsmodell för denna data? Motivera era svar.
2. Ge förslag på hur modellen skulle kunna förbättras utifrån era observationer i detta diagram.
3. Ange ett eller flera potentiella problem med att prediktera framtida temperaturer på detta sätt.

Kom ihåg att alla **tabeller och diagram** måste förses med **numrering och beskrivande text**. *Tips*: Använd `fig.cap` för automatisk numrering.