Il modello lineare

Introduzione

Consideriamo il problema di voler descrivere la relazione tra una quantità di interesse e un insieme di ulteriori variabili per le quali disponiamo di n osservazioni.

I modelli di regressione consentono di modellare la relazione tra una *variabile risposta*, usualmente indicata con y, e un insieme di variabili esplicative, $x_1, \ldots, x_p$ (inclusa l'intercetta); la struttura generale del modello è

$$\text{variabile risposta} \sim f(\cdot, \text{variabili esplicative})$$

dove $f(\cdot, \cdot)$ è una generica distribuzione di probabilità.

I *modelli lineari*, in particolare, sono un importante esempio di modelli statistici, che sono uno strumento fondamentale per l'analisi statistica dei dati, consentono di descrivere y come combinazione lineare di $x_1, \ldots, x_p$.

Consideriamo, ad esempio, i dati nel file *insulate.dat*, relativi al consumo di riscaldamento di un'abitazione prima e dopo un intervento di coibentazione. In particolare, i dati contengono le osservazioni della temperatura esterna media (in gradi Celsius) e del consumo settimanale di gas (in migliaia di piedi cubi) per 26 settimane prima e 30 settimane dopo l'intervento

```
insulate <- read.table('Insulate.dat', col.names=c("when","temp","cons"))
str(insulate)
```

```
## 'data.frame':    56 obs. of  3 variables:
##  $ when: chr  "prima" "prima" "prima" "prima" ...
##  $ temp: num  -0.8 -0.7 0.4 2.5 2.9 3.2 3.6 3.9 4.2 4.3 ...
##  $ cons: num  7.2 6.9 6.4 6 5.8 5.8 5.6 4.7 5.8 5.2 ...
```

```
insulate$when<-as.factor(insulate$when)
summary(insulate)
```

```
##     when         temp             cons
##  dopo :30   Min.   :-0.800   Min.   :1.300
##  prima:26   1st Qu.: 3.050   1st Qu.:3.500
##             Median : 4.900   Median :3.950
##             Mean   : 4.875   Mean   :4.071
##             3rd Qu.: 7.125   3rd Qu.:4.625
##             Max.   :10.200   Max.   :7.200
```

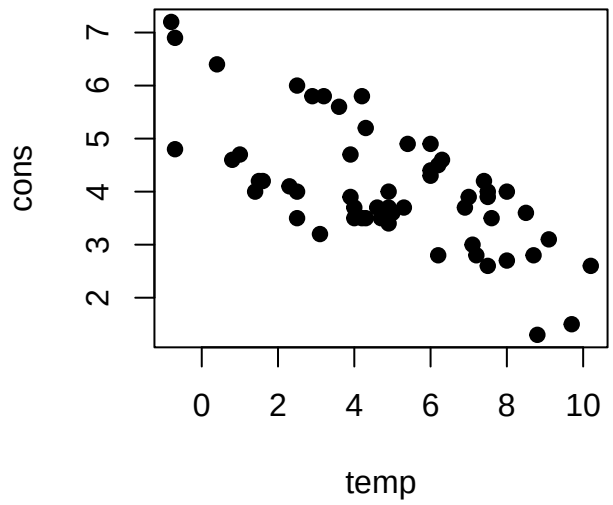
```
attach(insulate)
```

Regressione lineare semplice

Il nostro obiettivo è studiare la relazione tra il consumo (`cons`) e la temperatura esterna (`temp`) tenendo conto anche dell'effetto dell'intervento di isolamento (`when`). Per il momento tralasciamo questa ultima informazione.

Si indichi con (x_i, y_i), $i = 1, \ldots, 56$, l'insieme dei valori relativi alla temperatura (x) ed al consumo (y).

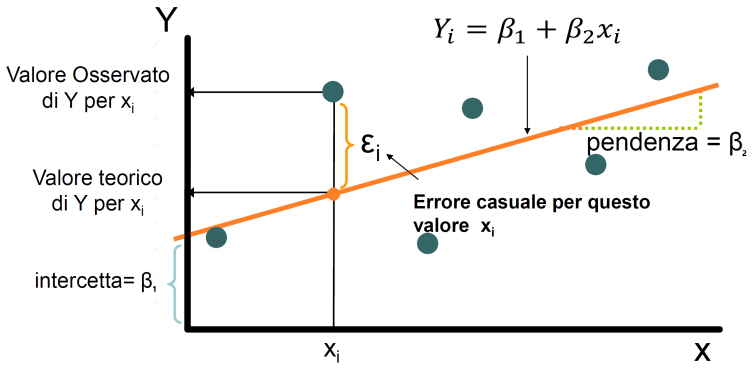
```
plot(temp, cons, pch=19)
```



Assumiamo che $y_1, \ldots, y_{56}$ siano realizzazioni di variabili aleatorie

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i$$

con ε_i variabili aleatorie indipendenti aventi distribuzione normale $N(0, \sigma^2)$, $i = 1, \ldots, 56$. L'errore è, quindi, una componente casuale con media 0 e varianza costante σ^2 e tiene conto del fatto che la relazione tra y e x non è esatta. Una ulteriore assunzione cruciale è quella che i termini di errore (e quindi le diverse osservazioni) siano indipendenti.



La media della variabile risposta Y data l'esplicativa x è data da $E(Y|x) = \beta_1 + \beta_2 x$; quindi $E(Y|x)$ aumenta di β_2 per ogni incremento unitario di x e l'intercetta, β_1, è la media di Y quando $x = 0$.

La funzione `lm` viene utilizzata per stimare un modello lineare, ottenendo le stime dei coefficienti β_1 e β_2, in questo caso, attraverso il *metodo dei minimi quadrati*, ovvero gli stimatori $\widehat{\beta}_1$ e $\widehat{\beta}_2$ minimizzano la funzione detta *Sum of Squares*

$$\sum_{i=1}^{n}(y_i - (\widehat{\beta}_1 + \widehat{\beta}_2\, x_i))^2$$

Se si adotta l'ipotesi di normalità, questi sono anche stimatori di massima verosimiglianza. I **residui** del modello sono i valori

$$\widehat{\varepsilon}_i = y_i - \widehat{y}_i,$$

dove $\widehat{y}_i = \widehat{\beta}_1 + \widehat{\beta}_2\, x_i$ sono i valori stimati. Lo stimatore della varianza dell'errore σ^2 è la varianza campionaria corretta dei residui:

$$s^2 = \frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{n-2}$$

```
ins.lm <- lm(cons ~ temp)
```

Nella formula, a sinistra del segno $\sim$ abbiamo la variabile risposta (o dipendente), mentre a destra c'è la variabile esplicativa (o indipendente). L'intercetta β_1 viene inclusa automaticamente nel modello. La funzione `lm()` produce un oggetto della classe `lm`:

```
class(ins.lm)
```

```
## [1] "lm"
```

```
length(ins.lm)
```

```
## [1] 12
```

```
names(ins.lm)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

`ins.lm` è un elenco di diversi tipi di output, tra cui le stime dei coefficienti e i residui del modello:

```
ins.lm$coefficients
```

```
## (Intercept)        temp
##   5.4861933  -0.2902082
```

```
ins.lm$residuals
```

```
##           1           2           3           4           5           6
##  1.48164017  1.21066099  1.02988996  1.23932707  1.15541033  1.24247278
##           7           8           9          10          11          12
##  1.15855604  0.34561848  1.53268093  0.96170174  0.98093071  1.15505560
##          13          14          15          16          17          18
##  0.55505560  0.65505560  0.81309723  0.94211804  0.21624293  0.44526375
##          19          20          21          22          23          24
##  0.86134701  0.69036782  0.59036782  0.21938864  0.83547190  0.58057597
##          25          26          27          28          29          30
##  0.25470086  0.07392983 -0.88933901 -0.65402678 -0.49598515 -1.07990189
##          31          32          33          34          35          36
## -0.85088108 -0.82186026 -0.71871456 -0.76067293 -1.26067293 -1.38654804
##          37          38          39          40          41          42
## -0.45438152 -0.82536070 -0.62536070 -0.76731907 -0.73829826 -0.45123581
##          43          44          45          46          47          48
## -0.62221500 -0.66417337 -0.36417337 -0.06417337 -0.43515255 -0.24809011
```

```
##          49          50          51          52          53          54
## -0.88690277 -0.42571544 -0.59669462 -0.70963218 -0.46452810 -0.16138240
##          55          56
## -1.63236158 -1.17117425
```

Altre funzioni di base come `coef()`, `fitted()` e `resid()` estraggono rispettivamente le stime, i valori stimati e i residui:

```
# not run
coef(ins.lm)
fitted(ins.lm)
resid(ins.lm)
```

Disponiamo poi di alcune funzioni per ottenere *intervalli di confidenza* sui parametri

```
confint(ins.lm)
```

```
##                  2.5 %     97.5 %
## (Intercept)  5.0136286  5.9587580
## temp        -0.3748234 -0.2055929
```

e una sintesi delle informazioni rilevanti sul modello stimato:

```
summary(ins.lm)
```

```
##
## Call:
## lm(formula = cons ~ temp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6324 -0.7119 -0.2047  0.8187  1.5327
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.4862     0.2357  23.275  < 2e-16 ***
## temp         -0.2902     0.0422  -6.876 6.55e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8606 on 54 degrees of freedom
## Multiple R-squared:  0.4668, Adjusted R-squared:  0.457
## F-statistic: 47.28 on 1 and 54 DF,  p-value: 6.545e-09
```

Il risultato di questo ultimo comando presenta:

- alcune statistiche riassuntive relative ai residui $\widehat{\varepsilon}_i$;
- la tabella dei coefficienti con le stime $\hat{\beta}_j$ (`Estimate`);
- gli standard errors (`Std. Error`) dati da $\sqrt{\hat{\mathrm{V}}(\hat{\beta}_j)}$;
- il valore osservato, t_j^{oss}, della statistica test $t_j = \hat{\beta}_j / \sqrt{\hat{\mathrm{V}}(\hat{\beta}_j)}$ per la verifica dell'ipotesi $H_0 : \beta_j = 0$ contro $H_1 : \beta_j \neq 0$ (t value);
- il valore-p (o p-value) calcolato come $\gamma = 2 \Pr_{H_0}(t_j > |t_j^{\text{oss}}|)$, indicato con `Pr(>|t|)`. Si noti che per l'interpretazione del p-value vengono riportati dei simboli per i quali è presente la legenda nella parte finale dell'output: $***$ indica che $\gamma < 0.001$ (forte significatività), $**$ indica che $0.001 < \gamma < 0.01$ (significatività abbastanza forte), e così via fino alla situazione in cui $\gamma > 0.10$ e quindi l'evidenza contro H_0 è molto debole, per cui l'ipotesi non viene rifiutata.

L'*errore standard dei residui* (`Residual standard error`) è la radice quadrata della stima corretta della

varianza σ^2 del termine di errore, insieme con i suoi gradi di libertà (pari a $n-2$, se n è il numero delle osservazioni).

Il *coefficiente di determinazione* R^2 e una sua versione corretta per il numero di coefficienti nel modello, R^2_c, sono indicati con `Multiple R-Squared` e `Adjusted R-Squared`, rispettivamente:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

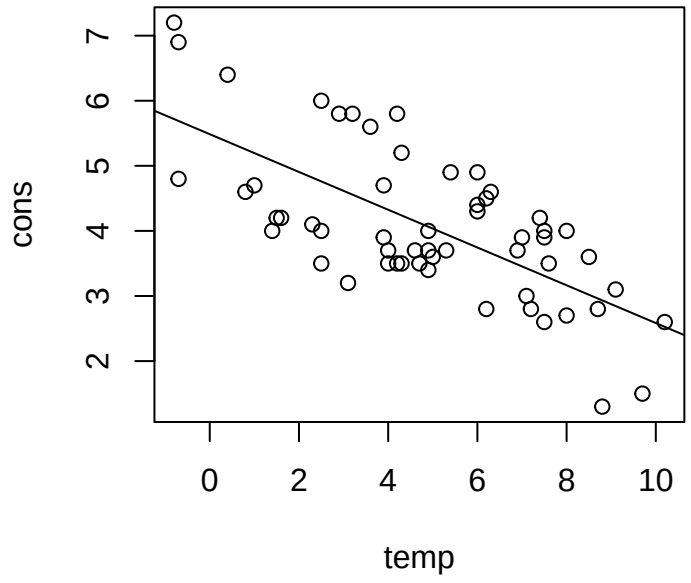
$$R^2_c = R^2 - \frac{p-1}{n-p}(1-R^2),$$

dove p è il numero di coefficienti presenti nel modello (in questo caso $p = 2$). L'indice R^2 (o R^2_c in presenza di più esplicative) è un valore tra 0 e 1 che rappresenta il rapporto tra la devianza di regressione e la devianza totale.

Infine, l'output riporta il valore della statistica F (`F-statistic`) per l'ipotesi di nullità di tutti i coefficienti, ad eccezione dell'intercetta, che fornisce informazioni sull'adeguatezza del modello. Nel modello di regressione lineare semplice tale test si riduce alla verifica di $H_0 : \beta_2 = 0$ contro $H_1 : \beta_2 \neq 0$, con $F \sim F_{1,n-2}$ (F di Fisher con 1 e $n-2$ gradi di libertà) sotto H_0.

Il modello stimato si può rappresentare nel seguente modo

```
plot(temp, cons)
abline(ins.lm)
```



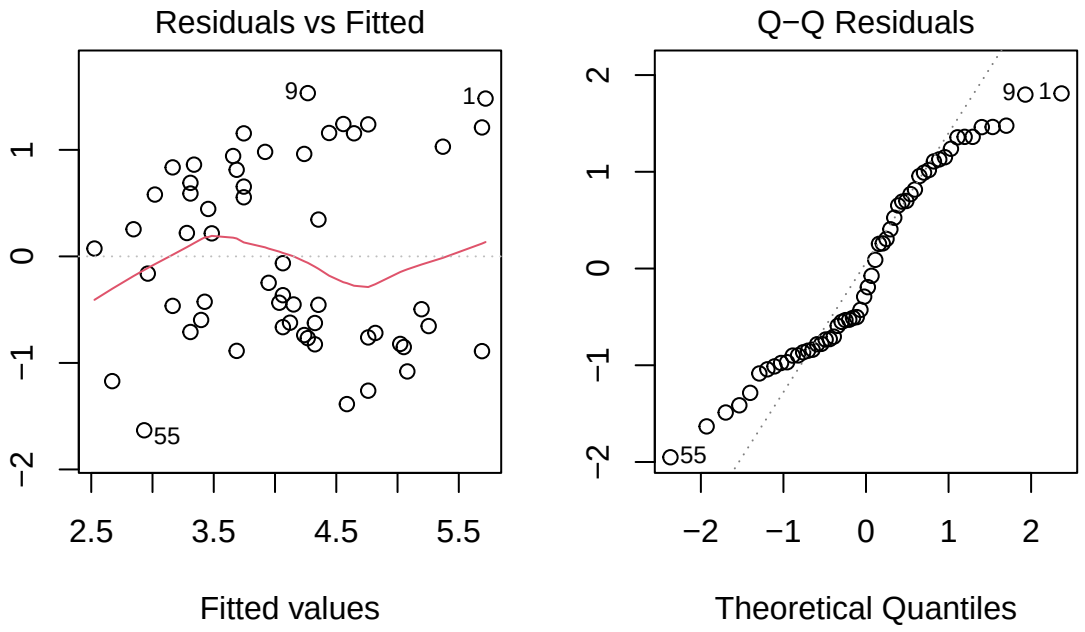
Nel modello lineare il coefficiente β_r ($r > 1$) rappresenta la variazione attesa nella risposta per una variazione unitaria dell'esplicativa x_r. Questo vale anche quando le esplicative sono più di una, come nel modello di regressione lineare multipla, dove ciascun coefficiente indica la variazione della variabile risposta per un incremento unitario della variabile esplicativa, a parità di altre condizioni.

La retta di regressione $\widehat{y}_i = 5.4861933 - 0.2902082 x_i$ indica che, all'aumentare della temperatura di un grado, il consumo diminuisce in media di circa $0.29 \times 1000 = 290$ piedi cubi, pari a circa 8m^3 (in generale, il valore

assoluto di $\widehat{\beta}_2$ indica di quanto varia la Y al variare di una unità della x). Il valore dell'intercetta è invece il valore di y corrispondente a $x = 0$.

L'**analisi dei residui** è importante per verificare la qualità del modello, ovvero se e fino a che punto le osservazioni sono compatibili con le ipotesi

```
par(mfrow=c(1,2), pty="s", mar=c(3,2,3,2))
plot(ins.lm, which = 1:2)
```



Se il modello di regressione lineare è adeguato, ci si attende che i residui si distribuiscano attorno ad una retta in modo casuale, senza mostrare tendenze evidenti. Se invece la rappresentazione grafica delle coppie $(x_i, \widehat{y}_i)$ rivela, ad esempio, una successione dapprima di valori positivi, poi di valori negativi e, infine, ancora una serie di valori positivi, il modello lineare appare inadeguato ed occorre passare a modelli alternativi (ad esempio, una funzione parabolica).

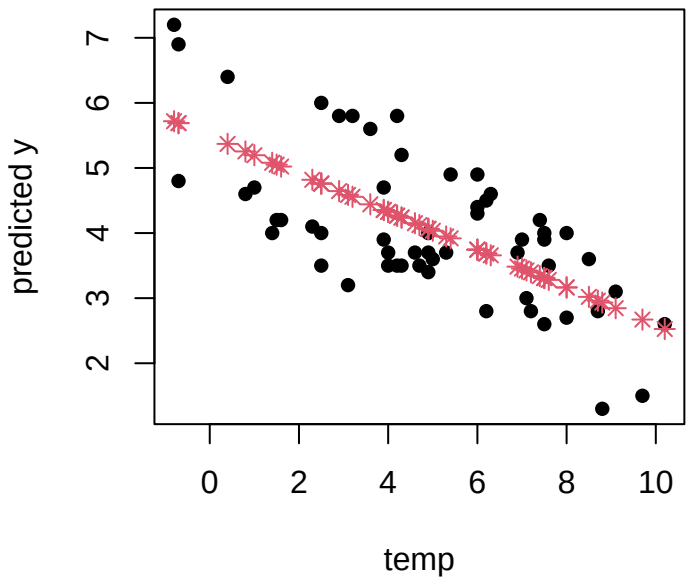
Stima della media e previsione

Un'altra funzione importante è la funzione `predict()` che, applicata ad un oggetto di classe `lm` può essere utilizzato per ottenere una stima della media di Y o una previsione in corrispondenza di un nuovo valore di Y:

```
ins.mean <- predict(ins.lm, interval="none")
head(ins.mean)
```

```
##        1        2        3        4        5        6
## 5.718360 5.689339 5.370110 4.760673 4.644590 4.557527
```

```
plot(temp, cons, pch=16, ylab="predicted y")
points(temp, ins.mean, pch=8, col=2)
```

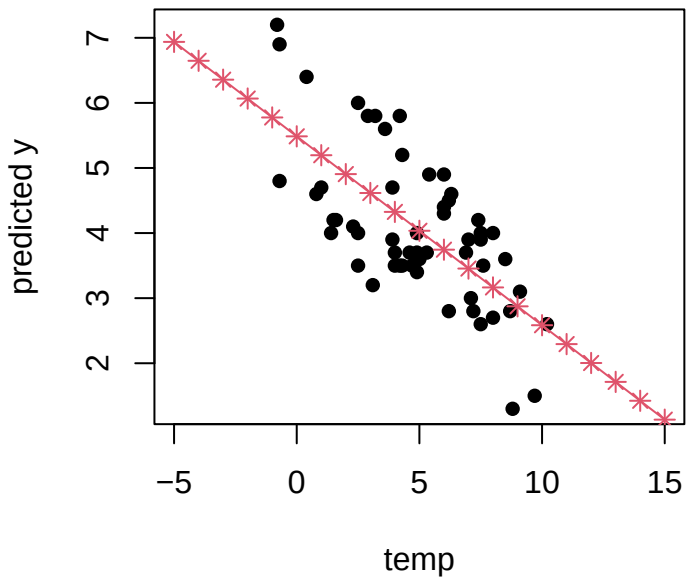


Possiamo generare dei nuovi valori della esplicativa e calcolare le previsioni per Y sulle nuove unità

```
new.data <- data.frame(temp=seq(-5, 15))
ins.meannew <- predict(ins.lm, newdata = new.data, interval="none")
head(ins.meannew)
```

```
##        1        2        3        4        5        6
## 6.937234 6.647026 6.356818 6.066610 5.776401 5.486193
```

```
plot(temp, cons, pch=16, xlim=range(new.data$temp), ylab="predicted y")
points(new.data$temp, ins.meannew, pch=8, col=2)
lines(new.data$temp, ins.meannew, col=2)
```

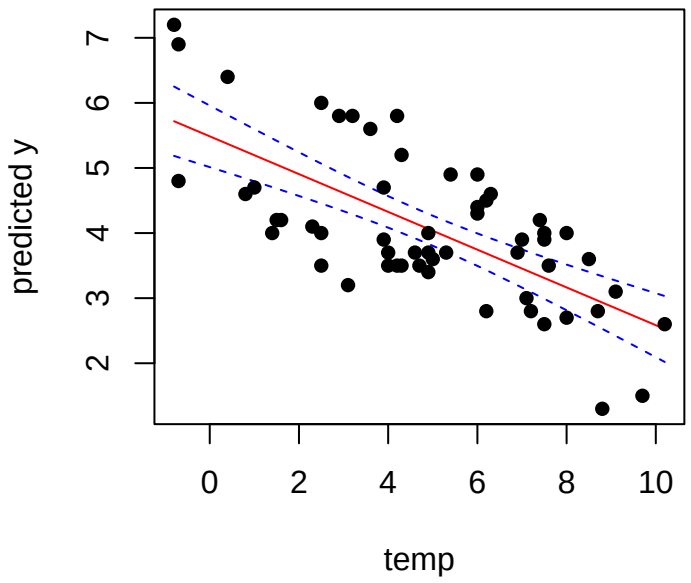


La funzione `predict()` consente di ottenere gli intervali di confidenza e/o previsione: i primi si ottengono impostando `interval="confidence"`:

```
ins.conf <- predict(ins.lm, interval="confidence")
head(ins.conf)
```

```
##        fit      lwr      upr
## 1 5.718360 5.185682 6.251038
## 2 5.689339 5.164276 6.214402
## 3 5.370110 4.926782 5.813438
## 4 4.760673 4.454818 5.066528
## 5 4.644590 4.359828 4.929351
## 6 4.557527 4.286881 4.828173
```

```
matplot(sort(temp), ins.conf[order(temp),], type="l", ylim=range(cons),
        lty=c("solid","dashed","dashed"), col=c("red","blue","blue"),
        xlab="temp", ylab="predicted y")
points(temp, cons, pch=16)
```



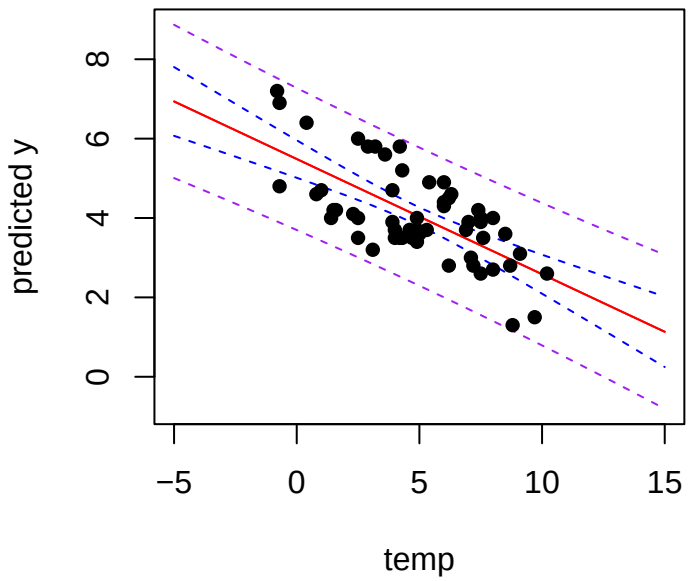
Un intervallo di confidenza riflette l'incertezza attorno ai valori medi di previsione, mentre un intervallo di previsione riflette l'incertezza attorno a un singolo valore, e sarà quindi più ampio di un intervallo di confidenza per lo stesso valore.

Gli intervali di previsione si possono ottenere impostando `interval="prediction"`

```
ins.confnew <- predict(ins.lm, newdata = new.data, interval="confidence")
ins.pred <- predict(ins.lm, newdata = new.data, interval="prediction")
```

Otteniamo il grafico con la retta di regressione in rosso, l'intervallo di previsione in viola e le bande di confidenza in blu:

```
matplot(new.data$temp, ins.pred, type="l", lty=c("solid","dashed","dashed"),
        col=c("red","purple","purple"), xlab="temp", ylab="predicted y")
matlines(new.data$temp, ins.confnew, type="l", lty=c("solid","dashed","dashed"),
         col=c("red","blue","blue"))
points(temp, cons, pch=16)
```



Modello con più esplicative

Il modello stimato nella sezione precedente ha evidenziato una relazione lineare negativa tra la temperatura esterna e il consumo di gas, come ci si poteva attendere. Possiamo estendere il modello per valutare se tale relazione possa subire l'effetto di un intervento di isolamento.

In generale, i modelli di interesse possono comprendere più variabili esplicative, di natura diversa. Il **modello di regressione multipla** ha lo scopo di spiegare l'effetto congiunto delle variabili esplicative su una variabile risposta e assume la forma

$$Y_i = \beta_1 + \beta_2 x_{i2} + \beta_3 x_{i3} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

per $i = 1, \ldots, n$ e $\varepsilon_i \sim iidN(0, \sigma^2)$.

Tornando all'esempio precedente, includiamo la variabile `when` nel modello iniziale. Poiché si tratta di una variabile qualitativa, tale variabile viene trasformata in una variabile indicatrice (o *dummy*):

$$z_i = \begin{cases} 1, & \texttt{when} = \texttt{prima} \\ 0, & \text{altrimenti} \end{cases}$$

per $i = 1, \ldots, 56$. Indichiamo quindi con (y_i, x_i, z_i), $i = 1, \ldots, 56$, i 56 valori di `cons`, `temp` e la variabile indicatrice per `when`. Scriviamo il modello come

$$Y_i = \beta_1 + \beta_2\, x_i + \beta_3\, z_i + \beta_4\, x_i\, z_i + \varepsilon_i$$

per $i = 1, \ldots, 56$, e $\varepsilon_i \sim N(0, \sigma^2)$ variabili aleatorie indipendenti. Il modello assume che, per $i = 1, \ldots, 26$ (prima dell'intervento) $Y_i \sim N((\beta_1 + \beta_3) + (\beta_2 + \beta_4)\, x_i, \sigma^2)$, mentre per $i = 27, \ldots, 56$ (dopo l'intervento) $Y_i \sim N(\beta_1 + \beta_2\, x_i, \sigma^2)$.

Si noti che per verificare se per la variabile `when` il valore 1 è associato a `prima` e 0 a `dopo` si può usare il comando

```
contrasts(when)
```

```
##       prima
## dopo      0
## prima     1
```

Il modello con interazione può essere stimato con

```
ins.lm.w <- lm(cons ~ temp*when)
# equivamlente a lm(cons ~ temp + when + temp:when)
```

e una sintesi dei risultati ottenuta nel modo seguente:

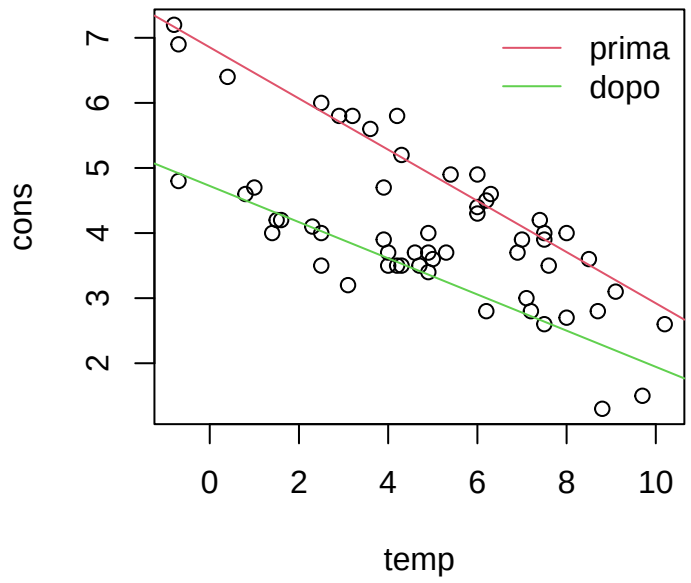
```
summary(ins.lm.w)
```

```
##
## Call:
## lm(formula = cons ~ temp * when)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.97802 -0.18011  0.03757  0.20930  0.63803
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     4.72385    0.11810  40.000  < 2e-16 ***
## temp           -0.27793    0.02292 -12.124  < 2e-16 ***
## whenprima       2.12998    0.18009  11.827 2.32e-16 ***
## temp:whenprima -0.11530    0.03211  -3.591 0.000731 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.323 on 52 degrees of freedom
## Multiple R-squared:  0.9277, Adjusted R-squared:  0.9235
## F-statistic: 222.3 on 3 and 52 DF,  p-value: < 2.2e-16
```

Tale modello equivale a stimare regressioni separate per ciascuna delle due categorie individuate dalla variabile dummy (prima e dopo).

L'intercetta e la pendenza per la categoria `prima` sono $\hat{\beta}_1+\hat{\beta}_3$ $(4.72385+2.12998)$ e $\hat{\beta}_2+\hat{\beta}_4$ $(-0.27793-0.11530)$, rispettivamente; mentre per la categoria `dopo` sono $\hat{\beta}_1$ (4.72385) e $\hat{\beta}_2$ (-0.27793), rispettivamente.

```
plot(temp, cons)
abline(6.85383, -0.39323, col=2)
abline(4.72385, -0.27793, col=3)
legend("topright", c("prima", "dopo"), col=2:3, lty=1, bty="n")
```



Per una temperatura esterna di 4 gradi Celsius, il valore atteso del consumo *dopo* dell'intervento è

$$\hat{y} = \hat{\beta}_1 + \hat{\beta}_2 * \text{temp} = 4.72385 - 0.27793 * 4 = 3.61213$$

il valore atteso del consumo *prima* dell'intervento è

$$\hat{y} = \hat{\beta}_1 + \hat{\beta}_3 + (\hat{\beta}_2 + \hat{\beta}_4) * \text{temp} = 4.72385 + 2.12998 + (-0.27793 - 0.11530) * 4 = 5.28091$$

Possiamo confrontare questo modello con quello precedente

$$Y_i = \beta_1 + \beta_2 \, x_i + \varepsilon_i,$$

dove $\varepsilon_i \sim N(0, \sigma^2)$, $i = 1, \ldots, 56$, usando il comando

```
anova(ins.lm,ins.lm.w)
```

```
## Analysis of Variance Table
##
## Model 1: cons ~ temp
## Model 2: cons ~ temp * when
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     54 39.995
## 2     52  5.425  2     34.57 165.67 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

$$\begin{array}{ll} H_0: & \beta_3 = \beta_4 = 0 \\ H_1: & \bar{H}_0 \end{array}$$

Il risultato conferma che vi è una differente relazione tra la temperatura e i consumi prima e dopo l'intervento, ovvero possiamo rifiutare l'ipotesi di uguaglianza dei consumi prima e dopo l'intervento, data la temperatura.

Otteniamo infine le stime/previsioni del consumo in base al nuovo modello, insieme ai rispettivi intervalli:

```
predict(ins.lm.w, newdata=data.frame(temp=c(-2,-2,10,10),
      when=c("prima","dopo","prima","dopo")),interval="confidence")
```

```
##        fit      lwr      upr
## 1 7.640305 7.285122 7.995488
## 2 5.279720 4.959716 5.599723
## 3 2.921439 2.676115 3.166764
## 4 1.944500 1.663660 2.225341
```

```
predict(ins.lm.w, newdata=data.frame(temp=c(-2,-2,10,10),
      when=c("prima","dopo","prima","dopo")),interval="prediction")
```

```
##        fit      lwr      upr
## 1 7.640305 6.901211 8.379399
## 2 5.279720 4.556873 6.002566
## 3 2.921439 2.228410 3.614469
## 4 1.944500 1.238117 2.650883
```

```
detach(insulate)
```