Business Statistics 41000

Multiple Linear Regression

Mladen Kolar

Multiple vs simple linear regression

Fundamental model is the same.

Basic concepts and techniques translate directly from SLR.

- Individual parameter inference and estimation is the same, conditional on the rest of the model.
- We still use `lm`, `summary`, `predict`, etc.

The hardest part would be moving to matrix algebra to translate all of our equations. Luckily, `R` does all that for you.

Polynomial regression

A nice bridge between SLR and MLR is polynomial regression.

Still only one X variable, but we add powers of X:

$$E[Y \mid X] = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_m X^m$$

You can fit any mean function if m is big enough.

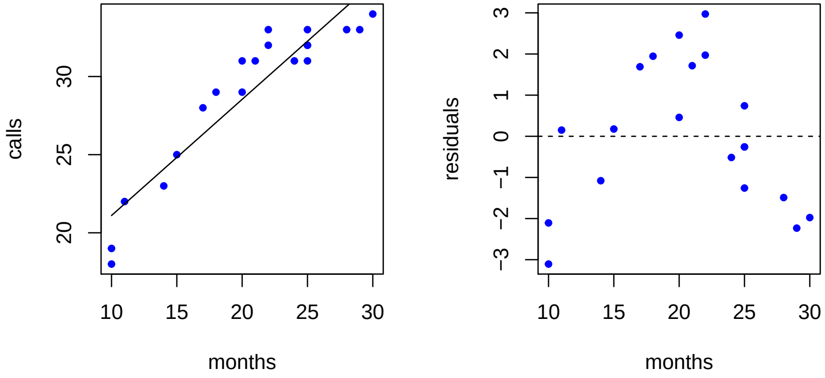
- Usually, $m = 2$ does the trick.

This is our first **multiple** linear regression!

Example: telemarketing/call-center data.

- How does length of employment (`months`) relate to productivity (number of `calls` placed per day)?

```
telemkt = read.csv("telemarketing.csv")
tele1 = lm(calls~months, telemkt)
xgrid = data.frame(months = 10:30)
par(mfrow=c(1,2))
plot(telemkt$months, telemkt$calls, pch=20, col=4,
     ylab="calls", xlab="months")
lines(xgrid$months, predict(tele1, newdata=xgrid))
plot(telemkt$months, resid(tele1), pch=20, col=4,
     ylab="residuals", xlab="months")
abline(h=0, lty=2)
```



It looks likethere is a polynomial shape to the residuals.

- We are leaving some predictability on the table
  . . . just not linear predictability.

Testing for nonlinearity

To see if you need more nonlinearity, try the regression which includes the next polynomial term, and see if it is significant.

For example, to see if you need a quadratic term,

- fit the model then run the regression $E[Y \mid X] = \beta_0 + \beta_1 X + \beta_2 X^2$.
- If your test implies $\beta_2 \neq 0$, you need X^2 in your model.

Test for a quadratic term:

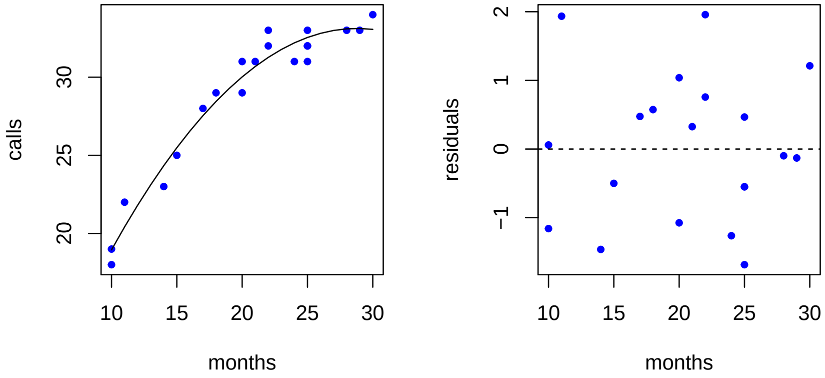
```
telemkt$months2 = telemkt$months^2
tele2 = lm(calls ~ months + months2, telemkt)
summary(tele2)  ## abbreviated output
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.14047    2.32263   -0.06     0.95
## months       2.31020    0.25012    9.24  4.9e-08 ***
## months2     -0.04012    0.00633   -6.33  7.5e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The quadratic $\texttt{months}^2$ term has a very significant z-value, so a better model is $\texttt{calls} = \beta_0 + \beta_1 \texttt{months} + \beta_2 \texttt{months}^2 + \varepsilon$.

Everything looks much better with the quadratic mean model.

```
xgrid = data.frame(months=10:30, months2=(10:30)^2)
par(mfrow=c(1,2))
plot(telemkt$months, telemkt$calls, pch=20, col=4,
     ylab="calls", xlab="months")
lines(xgrid$months, predict(tele2, newdata=xgrid))
plot(telemkt$months, rstudent(tele2), pch=20, col=4,
     ylab="residuals", xlab="months")
abline(h=0, lty=2)
```



Beyond SLR

Many problems involve more than one independent variable or factor which affects the dependent or response variable.

- Multi-factor asset pricing models (beyond CAPM).
- Demand for a product given prices of competing brands, advertising, household attributes, etc.
- More than size to predict house price!

In SLR, the conditional mean of Y depends on X. The multiple linear regression (MLR) model extends this idea to include more than one independent variable.

The MLR Model

The MLR model is same as always, but with more covariates.

$$Y \mid X_1, \ldots, X_d \sim N(\beta_0 + \beta_1 X_1 + \cdots + \beta_d X_d, \sigma^2)$$

Recall the key assumptions of our linear regression model:

(i) The conditional mean of Y is linear in the X_j variables.

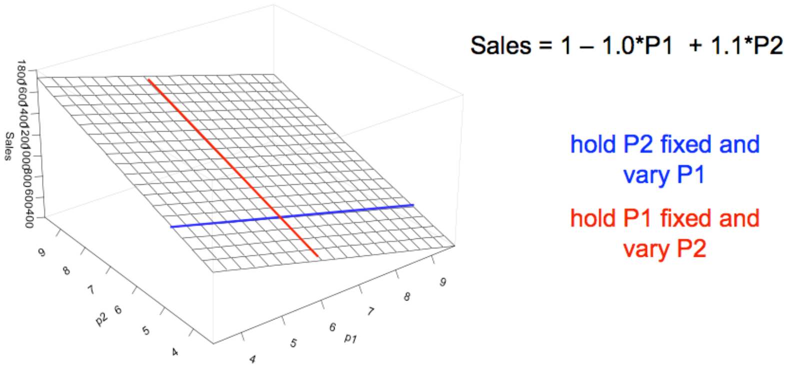
(ii) The additive errors (deviations from line)
- are Normally distributed
- independent from each other
- identically distributed (i.e., they have constant variance)

Our interpretation of regression coefficients can be extended from the simple single covariate regression case:

- Holding all other variables constant, β_j is the average change in Y per unit change in X_j.

If $d = 2$, we can plot the regression surface in 3D.

Consider `sales` of a product as predicted by price of this product (P1) and the price of a competing product (P2).



How do we estimate the MLR model parameters?

The principle of least squares is unchanged; define:

- fitted values $\hat{Y}_i = b_0 + b_1 X_{1i} + b_2 X_{2i} + \cdots + b_d X_{di}$
- residuals $e_i = Y_i - \hat{Y}_i$
- standard error $s = \sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n-p}}$, where $p = d + 1$.

Then find the best fitting plane, i.e., coefs $b_0, b_1, b_2, \ldots, b_d$, by minimizing the sum of squared residuals, s^2.

Obtaining these estimates in R is very easy:

```r
salesdata = read.csv("sales.csv")
(salesMLR = lm(Sales ~ P1 + P2, data = salesdata))
```

```
##
## Call:
## lm(formula = Sales ~ P1 + P2, data = salesdata)
##
## Coefficients:
## (Intercept)           P1           P2
##        1.00        -1.01         1.10
```

Residuals in MLR

As in the SLR model, the residuals in multiple regression are purged of any relationship to the independent variables.

We decompose Y into the part predicted by X and the part due to error.

$$Y = \hat{Y} + e$$

$$\text{corr}(X_j, e) = 0 \qquad \text{corr}(\hat{Y}, e) = 0$$

Standard errors

Conveniently, R's summary gives you all the standard errors.

```r
summary(salesMLR) ## abbreviated output
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.00269    0.00745     135   <2e-16 ***
## P1          -1.00590    0.00938    -107   <2e-16 ***
## P2           1.09787    0.00642     171   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0145 on 97 degrees of freedom
## Multiple R-squared:  0.998,  Adjusted R-squared:  0.998
## F-statistic: 2.39e+04 on 2 and 97 DF,  p-value: <2e-16
```

Inference for individual coefficients

Intervals and test statistics are exactly the same as in SLR.

- A $(1-\alpha)100\%$ C.I. for β_j is $b_j \pm z_{\alpha/2} s_{b_j}$.
- $z_{b_j} = (b_j - \beta_j^0)/s_{b_j} \sim N(0,1)$ is number of standard errors between the LS estimate and the null value.

Intervals/testing via b_j & s_{b_j} are one-at-a-time procedures:

- You are evaluating the $j^{\mathtt{th}}$ coefficient conditional on the other X's being in the model, but regardless of the values you've estimated for the other b's.

Categorical effects/dummy variables

To represent qualitative factors in multiple regression, we use dummy, binary, or indicator variables.

- temporal effects (1 if Holiday season, 0 if not)
- spatial (1 if in Midwest, 0 if not)

If a factor X takes R possible levels, we use $R-1$ dummies

- Allow the intercept to shift by taking on the value 0 or 1
- $\mathbb{1}_{[X=r]} = 1$ if $X = r$, 0 if $X \neq r$.

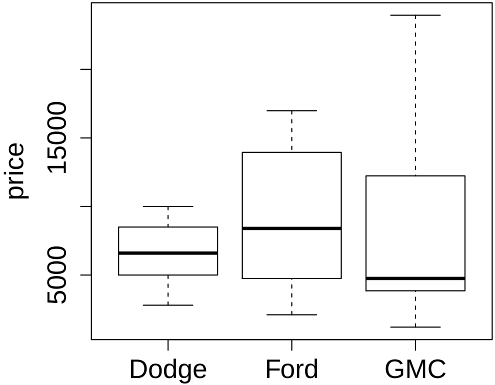
$$E[Y \mid X] = \beta_0 + \beta_1 \mathbb{1}_{[X=2]} + \beta_2 \mathbb{1}_{[X=3]} + \cdots + \beta_{R-1} \mathbb{1}_{[X=R]}$$

What is $E[Y \mid X = 1]$?

Example: back to the pickup truck data

Does `price` vary by `make`?

```
boxplot(price ~ make, data=pickup, ylab="price", cex.axis=1.5, cex.lab=1.5)
```



- GMC seems lower on average, but lots of overlap.
- Not much of a pattern.

Now fit with linear regression:

$$E[\texttt{price}|\texttt{make}] = \beta_0 + \beta_1 \mathbb{1}_{[\texttt{make}=\texttt{Ford}]} + \beta_2 \mathbb{1}_{[\texttt{make}=\texttt{GMC}]}$$

Easy in R (if make is a factor variable)

```r
summary(trucklm1 = lm(price ~ make, data=pickup))
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)     6554       1787    3.67  0.00067 ***
## makeFord        2314       2420    0.96  0.34439
## makeGMC         1442       2127    0.68  0.50150
```

The coefficient values correspond to our dummy variables.

What if you also want to include mileage?

- No problem.

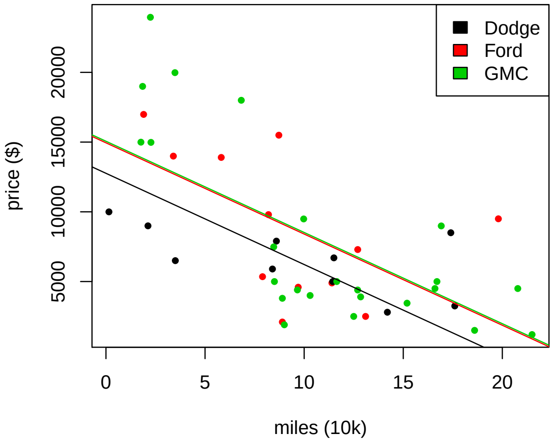
```r
summary(trucklm2 = lm(price ~ make + miles, data=pickup))
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  12761.8     1746.6   7.307 5.31e-09 ***
## makeFord      2185.7     1842.9   1.186    0.242
## makeGMC       2298.8     1627.0   1.413    0.165
## miles         -654.1      115.3  -5.671 1.18e-06 ***
```

All three brands expect to lose \$654 per 10k miles.

Different intercepts, same slope!

```
plot(pickup$miles, pickup$price, pch=20, col=pickup$make,
     xlab="miles (10k)", ylab="price ($)")
abline(a=coef(trucklm2)[1],b=coef(trucklm2)[4],col=1)
abline(a=(coef(trucklm2)[1]+coef(trucklm2)[2]),b=coef(trucklm2)[4],col=2)
abline(a=(coef(trucklm2)[1]+coef(trucklm2)[3]),b=coef(trucklm2)[4],col=3)
legend("topright", levels(pickup$make), fill=1:3)
```



Dodge trucks affect all slopes!

Variable interaction

So far we have considered the impact of each independent variable in a additive way.

We can extend this notion and include interaction effects through multiplicative terms.

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 (X_{1i} X_{2i}) + \cdots + \varepsilon_i$$

Interactions with dummy variables

Dummy variables separate out categories

- Different intercept for each category

Interactions with dummies separate out trends

- Different slope for each category

$$Y_i = \beta_0 + \beta_1 \mathbb{1}_{\{X_{1i}=1\}} + \beta_2 X_{2i} + \beta_3 (\mathbb{1}_{\{X_{1i}=1\}} X_{2i}) + \cdots + \varepsilon_i$$

When $X_1 = 0$, we have

$$E[Y \mid X_1 = 0, X_2] = \beta_0 + \beta_2 X_2$$

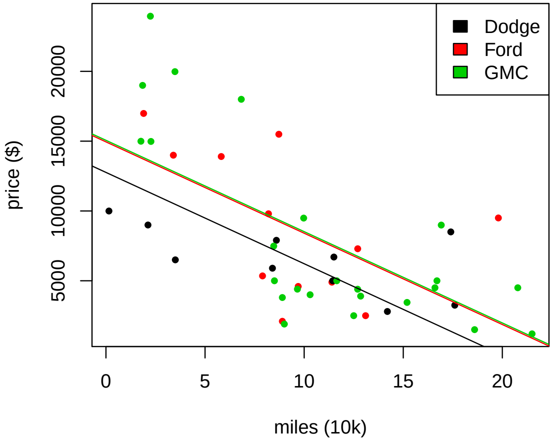
When $X_1 = 1$, we have

$$E[Y \mid X_1 = 1, X_2] = \beta_0 + \beta_1 + (\beta_2 + \beta_3) X_2$$

Same slope, different intercept

- Price difference does not depend on mileage!

```
trucklm2 = lm(price ~ make + mile, data=pickup)
```

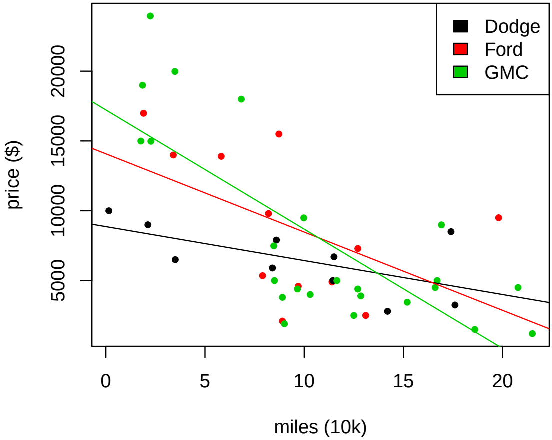


Dodge trucks affect all slopes!

Now add individual slopes!

- Price difference *varies* with miles!

```
trucklm3 = lm(price ~ make*miles, data=pickup)
```



Dodge doesn't effect Ford, GMC b_0, b_1

What do the numbers show?

```r
summary(trucklm3)
```

```
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)         8862       2508    3.53   0.0011 **
## makeFord            5216       3707    1.41   0.1671
## makeGMC             8360       3080    2.71   0.0097 **
## miles               -243        225   -1.08   0.2871
## makeFord:miles      -317        347   -0.91   0.3660
## makeGMC:miles       -611        268   -2.28   0.0282 *
```

```r
c(coef(trucklm3)[1], coef(trucklm3)[4]) ##(b_0,b_1) Dodge
```

```
## (Intercept)       miles
##        8862        -243
```

```r
c((coef(trucklm3)[1]+coef(trucklm3)[2]), ## b_0 Ford
  (coef(trucklm3)[4]+coef(trucklm3)[5])) ## b_1 Ford
```

```
## (Intercept)       miles
##       14079        -561
```

What do the numbers show?

```r
summary(trucklm3)
```

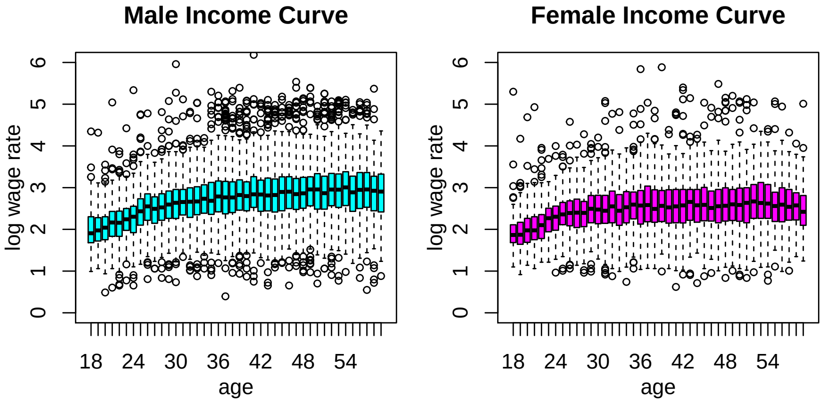
```
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)          8862       2508    3.53   0.0011 **
## makeFord             5216       3707    1.41   0.1671
## makeGMC              8360       3080    2.71   0.0097 **
## miles                -243        225   -1.08   0.2871
## makeFord:miles       -317        347   -0.91   0.3660
## makeGMC:miles        -611        268   -2.28   0.0282 *
```

```r
price.Ford = pickup$price[pickup$make=="Ford"]
miles.Ford = pickup$miles[pickup$make=="Ford"]
summary(lm(price.Ford ~ miles.Ford))
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    14079       3095    4.55   0.0011 **
## miles.Ford      -561        299   -1.87   0.0905 .
```

Case study in interaction

Use census data to explore the relationship between log wage rate (`log(income/hours)`) and age — a proxy for experience.

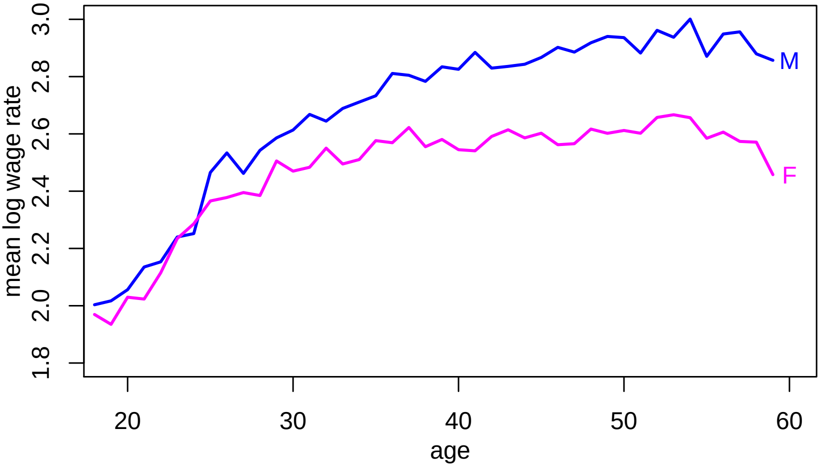


We look at people earning >$5000, working >500 hrs, and <60 years old.

A discrepancy between mean $\log(\mathrm{WR})$ for men and women.

- Female wages flatten at about 30, while men's keep rising.

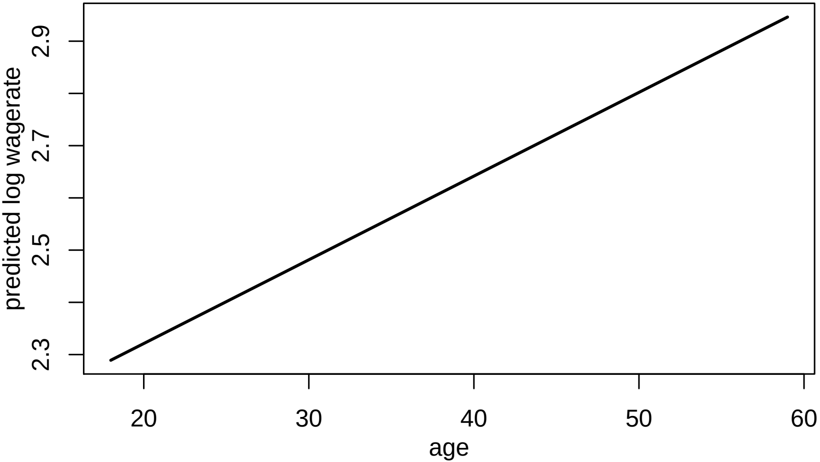
```r
men = sex == "M"
malemean = tapply(log.WR[men], age[men], mean)
femalemean = tapply(log.WR[!men], age[!men], mean)
```



The most simple model has

$$E[\log(\mathrm{WR})] = 2 + 0.016 \cdot \mathrm{age}.$$

```
wagereg1 = lm(log.WR ~ age)
```

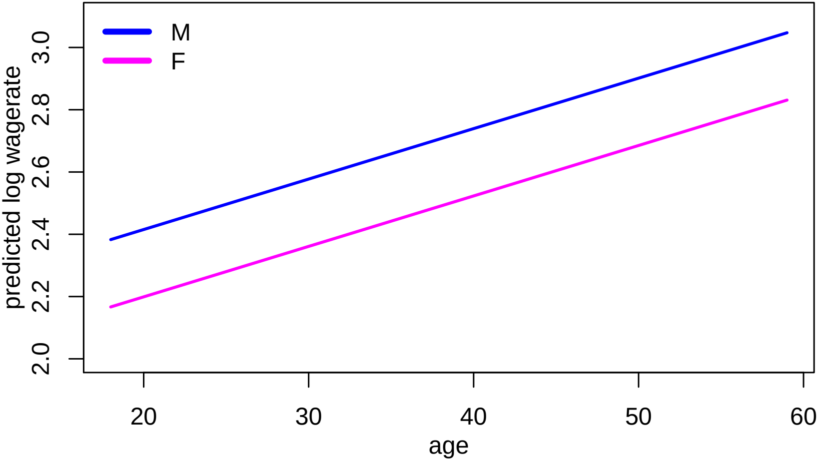


- You get one line for both men and women.

Add a sex effect with

$$E[\log(\mathrm{WR})] = 1.9 + 0.016 \cdot \mathrm{age} + 0.2 \cdot \mathbb{1}_{[\mathrm{sex}=M]}.$$

```r
wagereg2 = lm(log.WR ~ age + sex)
```

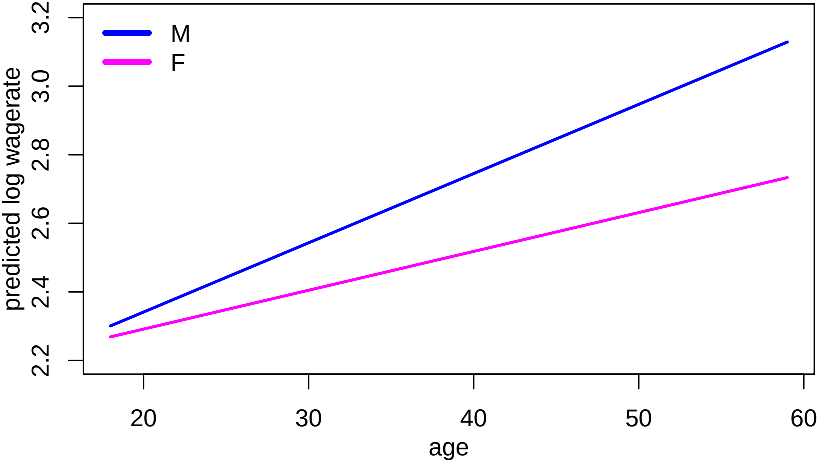


- The male wage line is shifted up from the female line.

With interactions

$$E[\log(\mathrm{WR})] = 2.1 + 0.011 \cdot \mathrm{age} + (-0.13 + 0.009 \cdot \mathrm{age})\mathbb{1}_{[\mathrm{sex}=M]}.$$

```r
wagereg3 = lm(log.WR ~ age*sex)
```

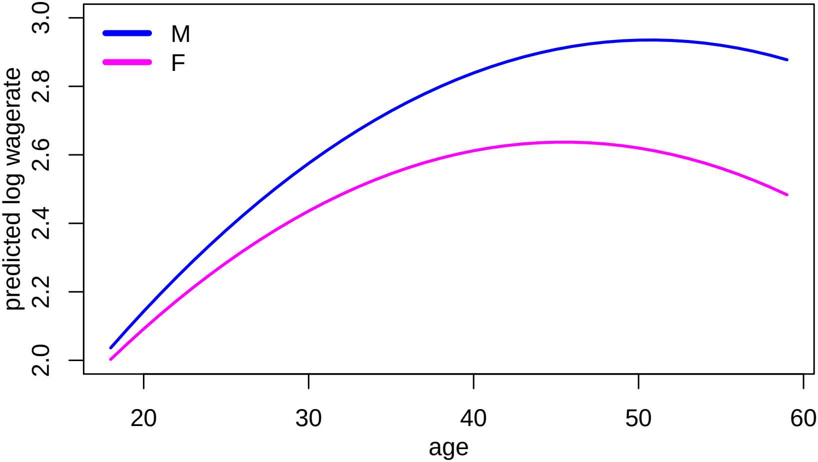


- The interaction term gives us different slopes for each sex.

& quadratics . . .

$$E[\log(\mathrm{WR})] = 0.9 + 0.077 \cdot \mathrm{age} - 0.0008 \cdot \mathrm{age}^2 + (-0.13 + 0.009 \cdot \mathrm{age})\mathbb{1}_{[\mathrm{sex}=M]}.$$

```r
wagereg4 = lm(log.WR ~ age*sex + age2)
```

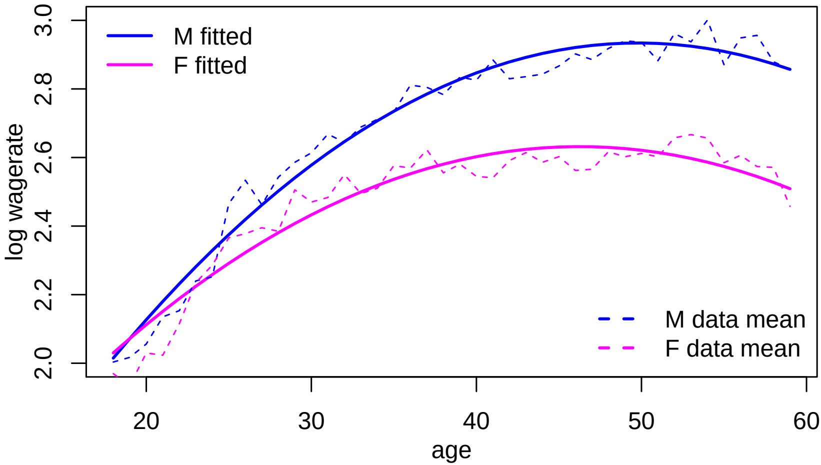


- age^2 allows us to capture a nonlinear wage curve.

Finally, add an interaction term on the curvature (age^2)

$E[\log(\text{WR})] = 1 + .07 \cdot \text{age} - .0008 \cdot \text{age}^2 + (.02 \cdot \text{age} - .00015 \cdot \text{age}^2 - .34)\mathbb{1}_{[\text{sex}=M]}$.

```
wagereg5 = lm(log.WR ~ age*sex + age2*sex)
```



- This full model provides a generally decent looking fit.