One-way Anova

All of the linear models we will discuss split the data vector Y into "signal" and "noise" components.

We will examine the results of the one-way ANOVA model from this perspective. First, we need to download the data.

Go to the course web page, then the *Notes and Handouts* section.

One-way Anova

All of the linear models we will discuss split the data vector Y into "signal" and "noise" components.

We will examine the results of the one-way ANOVA model from this perspective. First, we need to download the data.

Go to the course web page, then the *Notes and Handouts* section.

Right click on the *onewayanova_download.r* link and select *copy link location*

One-way Anova

All of the linear models we will discuss split the data vector Y into "signal" and "noise" components.

We will examine the results of the one-way ANOVA model from this perspective. First, we need to download the data.

Go to the course web page, then the *Notes and Handouts* section.

Right click on the *onewayanova_download.r* link and select *copy link location*

This should copy the URL for the Rdata file for this exercise. Use `setwd()` to switch to a directory you can write to.

One-way Anova

All of the linear models we will discuss split the data vector Y into "signal" and "noise" components.

We will examine the results of the one-way ANOVA model from this perspective. First, we need to download the data.

Go to the course web page, then the *Notes and Handouts* section.

Right click on the *onewayanova_download.r* link and select *copy link location*

This should copy the URL for the Rdata file for this exercise. Use `setwd()` to switch to a directory you can write to.

Paste the URL (now in the clipboard) between the quotes in the command `source("")`

One-way Anova

If all goes well, R should download three data frames, `onewayanova1-3`, to your workspace. Use `ls()` to verify that they are there.

One-way Anova

If all goes well, R should download three data frames, `onewayanova1-3`, to your workspace. Use `ls()` to verify that they are there.

Enter `plot(y∼x)` to graph the data.

One-way Anova

If all goes well, R should download three data frames, `onewayanova1-3`, to your workspace. Use `ls()` to verify that they are there.

Enter `plot(y~x)` to graph the data.

This should show three groups of measurements, indexed by the variable `x`

One-way Anova

If all goes well, R should download three data frames, `onewayanova1-3`, to your workspace. Use `ls()` to verify that they are there.

Enter `plot(y~x)` to graph the data.

This should show three groups of measurements, indexed by the variable x

You can get a boxplot by group with the command `boxplot(y~x)`

One-way Anova

If all goes well, R should download three data frames, `onewayanova1-3`, to your workspace. Use `ls()` to verify that they are there.

Enter `plot(y~x)` to graph the data.

This should show three groups of measurements, indexed by the variable x

You can get a boxplot by group with the command `boxplot(y~x)`

You summarize the data by group with the command

```
aggregate(y,
by=list(as.factor(onewayanova1$x)), summary)
```

One-way Anova

To run the ANOVA and save the results in a data structure called `aov1`, enter `aov1=aov(y~as.factor(x))`

One-way Anova

To run the ANOVA and save the results in a data structure called `aov1`, enter `aov1=aov(y~as.factor(x))`

The `aov` function produces an output structure very similar to the `lm` function that we used for regression.

Among other things, it contains the following numeric arrays:

- `fitted.values` contains what our model sees as the "signal" component of the data
- `residuals` contains the "noise" component corresponding to our estimated "signal"

One-way Anova

To run the ANOVA and save the results in a data structure called `aov1`, enter `aov1=aov(y~as.factor(x))`

The `aov` function produces an output structure very similar to the `lm` function that we used for regression.

Among other things, it contains the following numeric arrays:

- `fitted.values` contains what our model sees as the "signal" component of the data
- `residuals` contains the "noise" component corresponding to our estimated "signal"

If we added the "signal" and "noise" vectors term by term, we would recover the original y or data vector.

One-way Anova

To run the ANOVA and save the results in a data structure called `aov1`, enter `aov1=aov(y~as.factor(x))`

The `aov` function produces an output structure very similar to the `lm` function that we used for regression.

Among other things, it contains the following numeric arrays:

- `fitted.values` contains what our model sees as the "signal" component of the data
- `residuals` contains the "noise" component corresponding to our estimated "signal"

If we added the "signal" and "noise" vectors term by term, we would recover the original y or data vector.

One-way Anova

To run the ANOVA and save the results, enter

```
aov1=aov(y~as.factor(x))
```

One-way Anova

To run the ANOVA and save the results, enter
`aov1=aov(y~as.factor(x))`

The "signal" vector will be in `aov1$fitted.values`

The corresponding "noise" vector will be in
`aov1$residuals`

One-way Anova

To run the ANOVA and save the results, enter
`aov1=aov(y~as.factor(x))`

The "signal" vector will be in `aov1$fitted.values`

The corresponding "noise" vector will be in
`aov1$residuals`

Type `aov1$fitted.values` to display the "signal" component.

One-way Anova

To run the ANOVA and save the results, enter
`aov1=aov(y~as.factor(x))`

The "signal" vector will be in `aov1$fitted.values`

The corresponding "noise" vector will be in
`aov1$residuals`

Type `aov1$fitted.values` to display the "signal" component.

Notice that there are only three values (one for each group), and within a group every observation has the same "signal" value.

One-way Anova

To run the ANOVA and save the results, enter `aov1=aov(y~as.factor(x))`

The "signal" vector will be in `aov1$fitted.values`

The corresponding "noise" vector will be in `aov1$residuals`

Type `aov1$fitted.values` to display the "signal" component.

Notice that there are only three values (one for each group), and within a group every observation has the same "signal" value.

Now recall the data summary by group, `aggregate(y, by=list(as.factor(onewayanova1$x)), summary)`

One-way Anova

Notice that the group means (in the `x.Mean` column) match the fitted values for that group.

One-way Anova

Notice that the group means (in the `x.Mean` column) match the fitted values for that group.

This is characteristic of the one-way ANOVA model: the "signal" part of the model consists of the means at each level of the factor.

One-way Anova

Notice that the group means (in the `x.Mean` column) match the fitted values for that group.

This is characteristic of the one-way ANOVA model: the "signal" part of the model consists of the means at each level of the factor.

A good way to think of the "signal" is that it is the conditional expected value of y, given that the subject is in group x

One-way Anova

Notice that the group means (in the `x.Mean` column) match the fitted values for that group.

This is characteristic of the one-way ANOVA model: the "signal" part of the model consists of the means at each level of the factor.

A good way to think of the "signal" is that it is the conditional expected value of y, given that the subject is in group x

Recall that in the output for the regression model (from `lm`), we could see the coefficients of the linear model, but with `aov` we don't see them. It is customary not to show the coefficients for an ANOVA model, because they can be very confusing.

One-way Anova

Nonetheless, `aov` does produce estimated coefficients and it is instructive to examine them.

One-way Anova

Nonetheless, `aov` does produce estimated coefficients and it is instructive to examine them.

First recall that our linear model for the one-way ANOVA with three levels of the factor is:

$$y_i = \mu + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + e_i$$

One-way Anova

Nonetheless, aov does produce estimated coefficients and it is instructive to examine them.

First recall that our linear model for the one-way ANOVA with three levels of the factor is:

$$y_i = \mu + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + e_i$$

Remember that the x_{1i} value is one if this subject is in group 1, and zero if they are not. x_{2i} and x_{3i} are definined in similar fashion.

One-way Anova

Here's the confusing part. If we display the coefficients by entering `aov1$coefficients`, there are coefficients for the intercept, for x_{2i}, and for x_{3i}, but not for x_{1i}. Why is x_{1i} missing?

One-way Anova

Here's the confusing part. If we display the coefficients by entering `aov1$coefficients`, there are coefficients for the intercept, for x_{2i}, and for x_{3i}, but not for x_{1i}. Why is x_{1i} missing?

We can see the reason if we think of this as a linear algebra problem. The "ordinary least squares" techniques are sometimes called **projection** methods because the estimated "signal" is, geometrically speaking, the projection of y into the vector space spanned by the columns of the design matrix.

One-way Anova

Here's the confusing part. If we display the coefficients by entering `aov1$coefficients`, there are coefficients for the intercept, for x_{2i}, and for x_{3i}, but not for x_{1i}. Why is x_{1i} missing?

We can see the reason if we think of this as a linear algebra problem. The "ordinary least squares" techniques are sometimes called **projection** methods because the estimated "signal" is, geometrically speaking, the projection of y into the vector space spanned by the columns of the design matrix.

Things get confusing if the columns of the design matrix are not linearly independent. In this case, we must choose a linearly independent set or **basis** for the space spanned by the columns of x.

One-way Anova

The software does this for us automatically, so we don't have to give much thought to it, but it is crucial to understanding what the coefficients mean.

One-way Anova

The software does this for us automatically, so we don't have to give much thought to it, but it is crucial to understanding what the coefficients mean.

In the one-way ANOVA with three levels of the factor, the design matrix has four columns, one for μ which is all ones, and one for each of x_{1i}, x_{2i}, and x_{3i}. However, there are only three linearly independent columns.

One-way Anova

The software does this for us automatically, so we don't have to give much thought to it, but it is crucial to understanding what the coefficients mean.

In the one-way ANOVA with three levels of the factor, the design matrix has four columns, one for μ which is all ones, and one for each of x_{1i}, x_{2i}, and x_{3i}. However, there are only three linearly independent columns.

In this example, we can choose a basis for the design matrix X by discarding any one of the four columns. The estimated "signal" will be the same regardless of which one we discard.

One-way Anova

The software does this for us automatically, so we don't have to give much thought to it, but it is crucial to understanding what the coefficients mean.

In the one-way ANOVA with three levels of the factor, the design matrix has four columns, one for μ which is all ones, and one for each of x_{1i}, x_{2i}, and x_{3i}. However, there are only three linearly independent columns.

In this example, we can choose a basis for the design matrix X by discarding any one of the four columns. The estimated "signal" will be the same regardless of which one we discard.

In this case, the software decided to discard the column associated with group 1.

One-way Anova

With the column for group 1 discarded, the fitted values for the three groups are:

- Group 1: $E(Y) = \mu$
- Group 2: $E(Y) = \mu + \beta_2$
- Group 3: $E(Y) = \mu + \beta_3$

One-way Anova

With the column for group 1 discarded, the fitted values for the three groups are:

- Group 1: $E(Y) = \mu$
- Group 2: $E(Y) = \mu + \beta_2$
- Group 3: $E(Y) = \mu + \beta_3$

We can display the estimates of μ, β_2, and β_3 by entering `aov1$coefficients`

One-way Anova

With the column for group 1 discarded, the fitted values for the three groups are:

- Group 1: $E(Y) = \mu$
- Group 2: $E(Y) = \mu + \beta_2$
- Group 3: $E(Y) = \mu + \beta_3$

We can display the estimates of μ, β_2, and β_3 by entering `aov1$coefficients`

Notice that the corresponding "signal" value for each group, which is always $E(Y)$ for that group, matches the group means we obtained with `aggregate(y, by=list(as.factor(onewayanova1$x)), summary)`

One-way Anova

With the column for group 1 discarded, the fitted values for the three groups are:

- Group 1: $E(Y) = \mu$
- Group 2: $E(Y) = \mu + \beta_2$
- Group 3: $E(Y) = \mu + \beta_3$

We can display the estimates of μ, β_2, and β_3 by entering `aov1$coefficients`

Notice that the corresponding "signal" value for each group, which is always $E(Y)$ for that group, matches the group means we obtained with `aggregate(y, by=list(as.factor(onewayanova1$x)), summary)`

This will always be the case for the one-way ANOVA regardless of which column we discard to get our basis.

One-way Anova

We will look at one more example to see what happens if we choose a different basis. We can force R to discard the first column of the design matrix, the column of all ones corresponding to the coefficient μ, by specifying a "no intercept" model.

One-way Anova

We will look at one more example to see what happens if we choose a different basis. We can force R to discard the first column of the design matrix, the column of all ones corresponding to the coefficient μ, by specifying a "no intercept" model.

The R command for this is:

```
aovni=aov(0+as.factor(x))
```

One-way Anova

We will look at one more example to see what happens if we choose a different basis. We can force R to discard the first column of the design matrix, the column of all ones corresponding to the coefficient μ, by specifying a "no intercept" model.

The R command for this is:
`aovni=aov(0+as.factor(x))`

Now list the coefficients of the model with `aovni$coefficients` and notice that now they correspond to the three levels of the factor and match the group means.

One-way Anova

We will look at one more example to see what happens if we choose a different basis. We can force R to discard the first column of the design matrix, the column of all ones corresponding to the coefficient μ, by specifying a "no intercept" model.

The R command for this is:
`aovni=aov(0+as.factor(x))`

Now list the coefficients of the model with `aovni$coefficients` and notice that now they correspond to the three levels of the factor and match the group means.

Finally, enter `aovni$fitted.values` to confirm that the "signal" part of the model is still the mean for each group.