Week 2 Workshop

Multiple Linear Regression Pt 1 Quantitative predictors

This week we have started to learn about multiple linear regression. Multiple linear regression can be used to model numerical response variables using 2 or more explanatory variables. The explanatory variables can include numerical or categorical predictors and polynomial or interaction terms. This week we will focus on fitting main effects models using quantitative predictors. Next week, we will extend these concepts with interaction terms and qualitative predictors and in week 4 we will introduce polynomial terms.

The objective of this week's workshop is to:

- Learn how to fit a Multiple linear regression in R using `lm()`.
- Interpret relevant output from a Multiple Linear Regression using `anova()`, `summary()`, and `confint()`.
- Check conditions of a Multiple Linear Regression using `resid_panel()`.
- Introduce you to the `GGally` package as a method of exploratory analysis.

1 Example: Perch

The `Weight` (g), `Length` (cm) and `Width` (cm) of a sample of 56 perch were recorded. Also contained in the data set are the length class (`L`) and weight class (`Wd`) of the perch as well as the observation number (`Obs`). For this example today, we are just going to use `Length`, `Width` and `Weight`. Your task today is to **find a multiple linear regression model for `Weight` based on the measurements of `Length` and `Width`.**

Source: STAT2: Building Models for a World of Data, A. R. Cannon (2013), W. H. Freeman [Ed.].

1.1 Initial settings and importing data

Make sure you have your STAT210 or STAT410 project open and start a new R script. Name your new script something useful. EG Workshop 2 MLR. Use the `options()` command from last week to set the number of significant figures and hide the significance stars in outputs. Finally, load the `Perch.csv` data file.

▶ **CHECK**

You can view your data set by clicking on the name of the data frame (`per.df`), which appears under Data in the top right hand window of RStudio. Note the names and order of the variables. The first variable is simply the observation number and is not included in the analysis. The second variable is the response, `Weight`.

1.2 Exploratory Analysis

This week we will introduce you to a new command, `ggpairs()` in the `GGally` package. A pairs plot generates a matrix of scatter plots for every combination of 2 variables, which is especially useful when looking at multiple predictors as we do for Multiple Linear Regression. The `ggpairs()` command also provides some extra information such as correlation coefficients (r) between predictors and a density plot for each predictor.

Produce a pairs plot using the ggpairs() command which is contained within the GGally package.

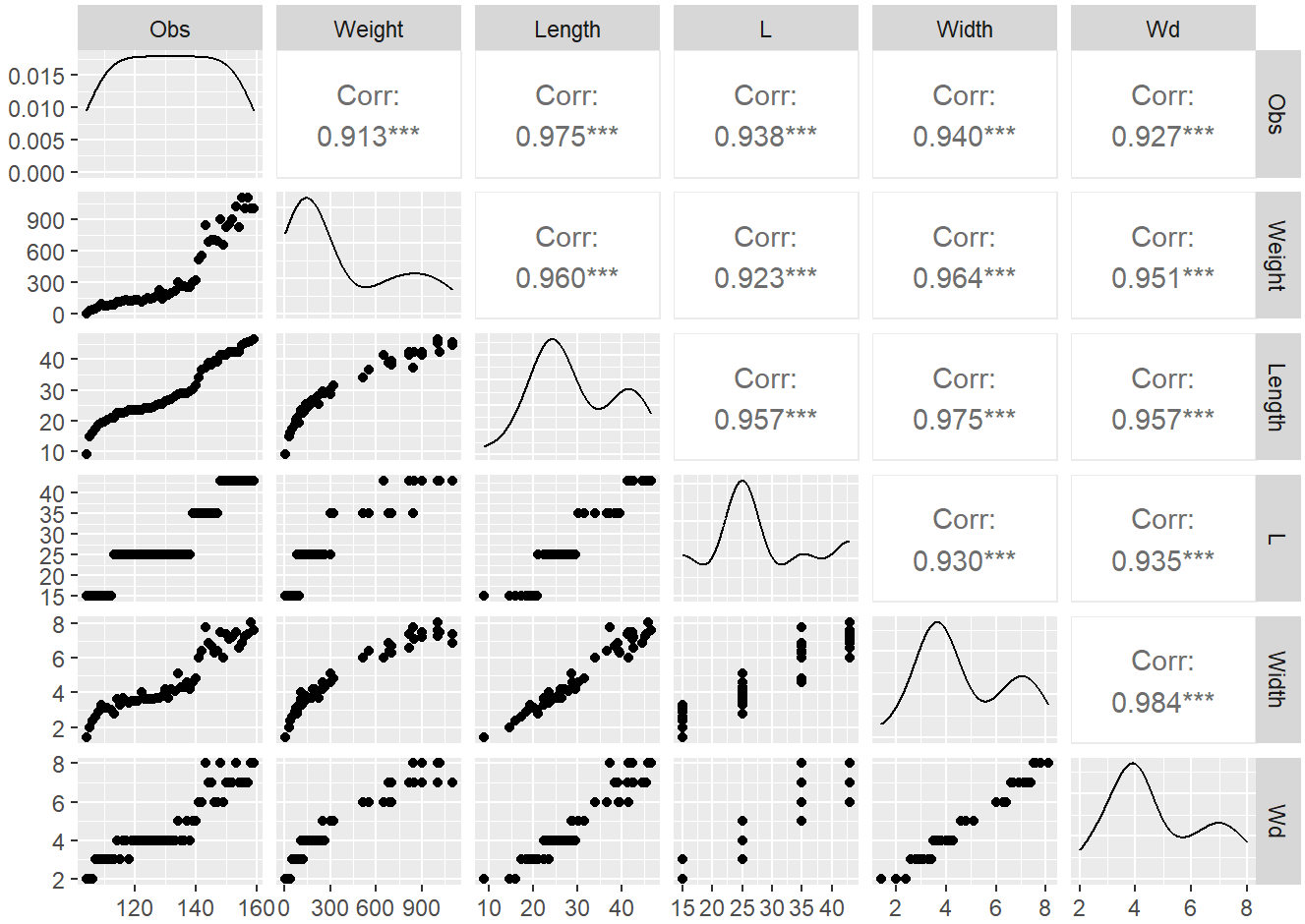
NOTE: You will need to install and load the `GGally` package in order to use `ggpairs`.

▶ **HINT**

```
ggpairs(per.df)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
## Warning in geom_point(): All aesthetics have length 1, but the data has 36 rows.
## ▫ Please consider using `annotate()` or provide this layer with data containing
##   a single row.
```

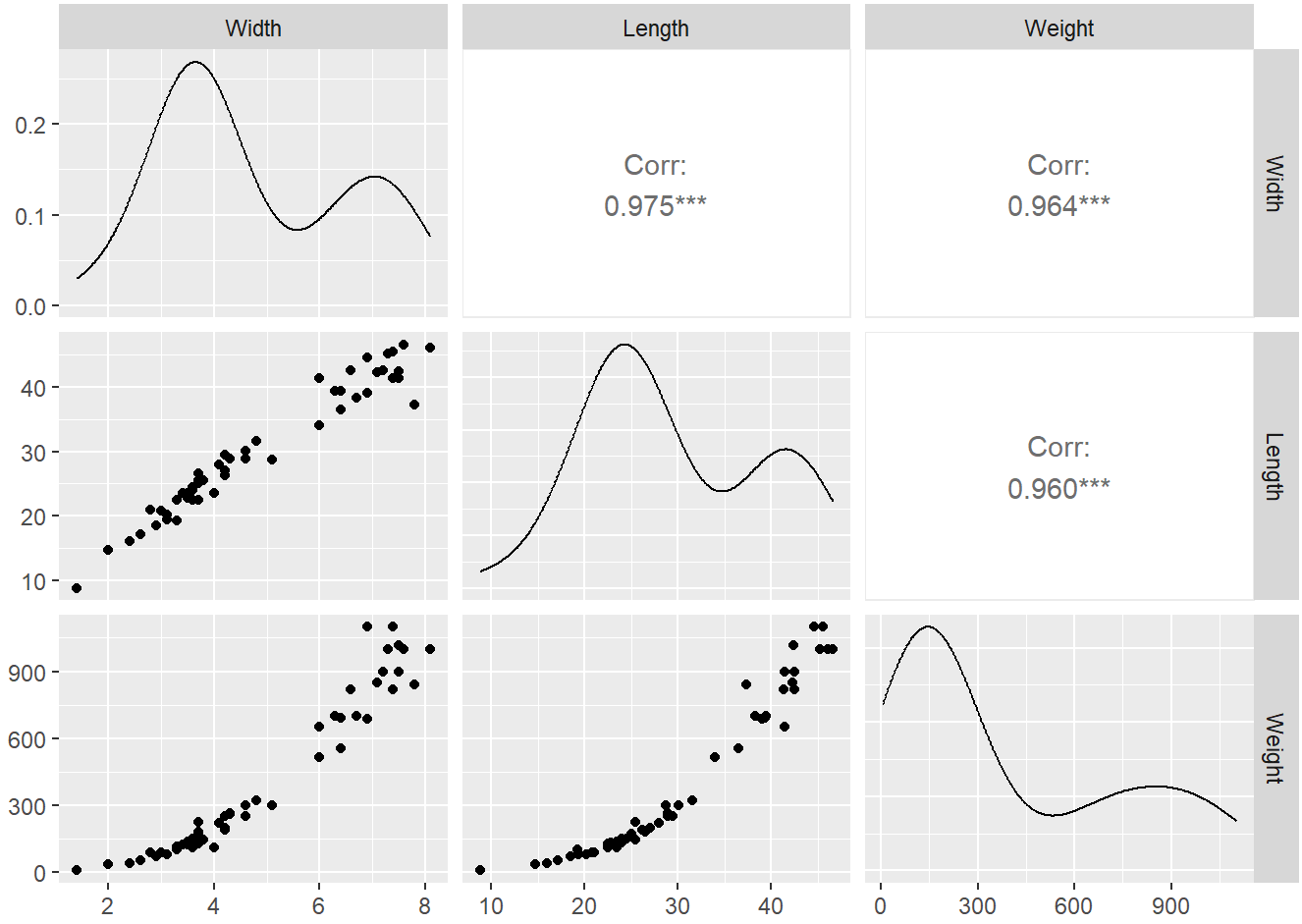


Just including the data name in the command will generate a pairs plot for all variables (columns) in your data frame. Sometimes data will contain variables that we are not interested in, as is the case here. For our example, we only want to include `Width`, `Length` and `Weight`. We can just include the variables we are interested in by specifying them with:

```
ggpairs(per.df, columns = c("Width", "Length", "Weight"))
```

OR we can use the column number in the data set instead. Here `Width` is the 5th column, so we use 5; `Length` is the 3rd column, so we use 3; and `Weight` is the 2nd column, so we use 2.

```
ggpairs(per.df, columns = c(5, 3, 2))
```



The order that you state the variables in the code does not need to match the order they are in the data frame. Whichever order you place the variable names, is the order that they will appear in the plot matrix. I like to place my response variable at the very end as it makes reading the relationships between the predictors and the response variable easier. You could place it anywhere in the order though.

Now that we have our exploratory plot, we should discuss the relationships between each of our predictors and the response variable and also any relationships between our predictors. Take a moment to **write down the relationships between the response and the predictors and between the two predictors.**

▶ **CHECK**

NOTE: A pairs plot is often a good starting place for looking at relationships, but keep in mind that a single well designed scatter plot (or some other type of plot like a boxplot if dealing with qualitative predictors as well) might be a better choice for publication or understanding interactions (we will talk about interactions in more detail in week 3 & 4).

Practice using `ggplot()`

Produce a plot looking at the relationship between `Weight` and `Length` using `ggplot()`. You will need to load `ggplot2` using the same `library()` command as you did for `GGally`.

NOTE: You should have installed `ggplot2` in last weeks workshop. If you haven't, you will need to do this first.

▶ **HINT**

1.3 Fit a main effects MLR model and look at output

Now create a main effects model for `Weight` (call it `mod1`) with `Length` and `Width` as the predictors. The code is similar to what we used in SLR last week, but now we need to add an extra variable to the code like so:

```
mod1<-lm(Weight~Width+Length, data=per.df)
```

NOTE: `Weight` as our response variable is still placed to the left of the `~` and our two predictors are placed on the right, separated with a `+` (like you would separate them in the equation).

To look at the output from our model, we can produce an ANOVA table with `anova()`, look at the `summary()` table of the regression coefficients and generate confidence intervals of the coefficients using `confint()`.

The p-values in the `anova()` table provide p-values when comparing increasingly complex models through the addition of an extra predictor. So these p-values indicate if the model with an additional predictor is useful when all predictors above in the sequence have been fitted first. This means that order of fit for the predictors is important when looking at this output. In our example we have fitted `Width` first and `Length` second. If we were to reverse the order of our predictors (EG `Weight~Length+Width`), the p-values in the anova table will change.

```
anova(mod1)
```

```
## Analysis of Variance Table
##
## Response: Weight
##           Df  Sum Sq Mean Sq  F value  Pr(>F)
## Width      1 6179065 6179065 785.7974 < 2e-16 ***
## Length     1   50267   50267   6.3925 0.01447 *
## Residuals 53  416762    7863
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Looking at the ANOVA table, we are comparing 3 models:

1. An intercept only model. This is always the default starting model in an ANOVA and is essentially just fitting a single mean for all values of Weight. EG `Weight` does not change in response to predictors.

   $\hat{Weight} = b_0$

2. A model with only `Width` fitted (EG a Simple linear regression). EG `Weight` changes in response to `Width` only.

   $\hat{Weight} = b_0 + b_1(Width)$

3. A model with `Width` and `Length` fitted (A Multiple linear regression). EG `Weight` changes in response to `Width` and `Length`.

$$\hat{y} = b_0 + b_1(Weight) + b_2(Length)$$

The p-value for `Width` in the ANOVA table is comparing the intercept only model to the model with only `Width` fitted. So here we are asking "Does adding `Width` to the model explain more variation in our response than an intercept only model?" Since the p-value is <0.05, we can say yes it does *OR* that a model with `Width` is a better model to explain the variation in `Weight` than an intercept (mean) only model.

The same concept can be applied to `Length` in the ANOVA output, only here we are comparing the model with only `Width` fitted to a model with `Width` and `Length` fitted. Since the p-value is less than 0.05, we can say that the more complex model;

$$\hat{Weight} = b_0 + b_1(Width) + b_2(Length)$$

explains significantly more variation in `Weight` than the simpler model;

$$\hat{Weight} = b_0 + b_1(Width)$$

The p-values in the `summary()` table indicate if the predictor is important when all other predictors are included in the model we fitted. Here we are looking at the individual components of the model we fitted, rather than comparing increasingly complex models as we were with the ANOVA table. This means that when interpreting the p-values from the summary output, order of fit is not important and changing the order of the predictors in our model will not change the p-values.

```
summary(mod1)
```

```
##
## Call:
## lm(formula = Weight ~ Width + Length, data = per.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -113.86  -59.02  -23.29   30.93  299.85
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -578.758     43.667 -13.254  < 2e-16 ***
## Width        113.500     30.265   3.750 0.000439 ***
## Length        14.307      5.659   2.528 0.014475 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 88.68 on 53 degrees of freedom
## Multiple R-squared:  0.9373, Adjusted R-squared:  0.9349
## F-statistic: 396.1 on 2 and 53 DF,  p-value: < 2.2e-16
```

Lets have a look at the output in the summary table a little more closely.

We can use the table of coefficients from the `summary()` output to **write the fitted model equation.** A fitted MLR model is defined as:

$$\hat{y} = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_k x_k$$

Where; b_0 is the intercept, b_1 is the coefficient for the first predictor x_1 (`Width`) and b_2 is the coefficient for the second predictor x_2 (`Length`). We can have parameters in our model up to the kth parameter. However, since we only have the two for this model, we will stop at x_2 for our equation. Therefore the equation for Weight using our fitted model is defined as:

$$\hat{Weight} = -578.76 + 113.50(Width) + 14.31(Length)$$

Finally, we can look at the the confidence intervals for each of the components of the model. Remember that the interpretation of the confidence interval for each predictor is made while assuming that the other predictors in the model are not changing.

```
confint(mod1)
```

```
##                   2.5 %     97.5 %
## (Intercept) -666.343179 -491.17237
## Width         52.796272  174.20304
## Length         2.957273   25.65749
```

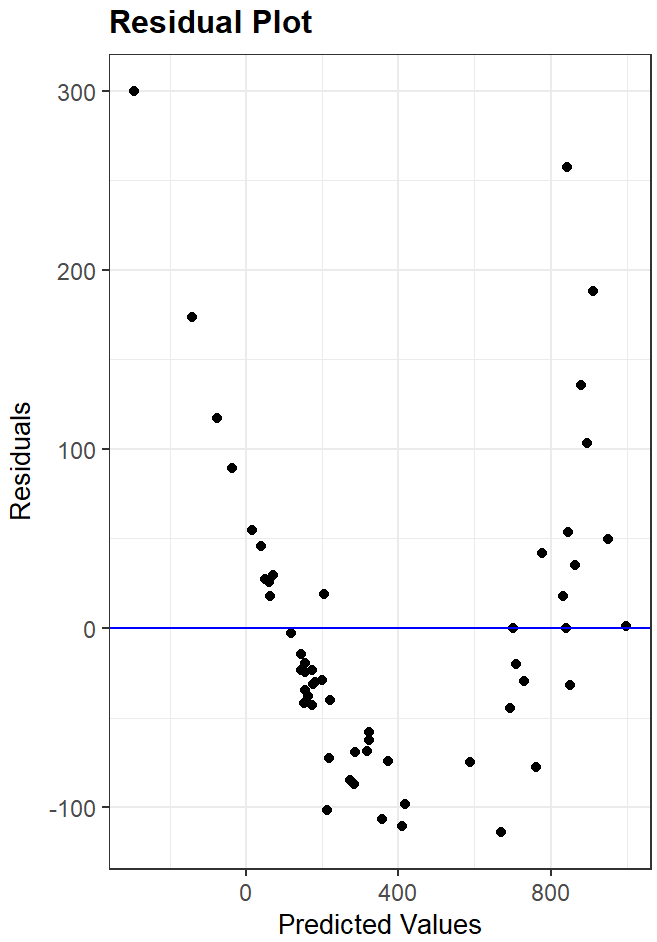
1.4 Check the Assumptions of the main effects model

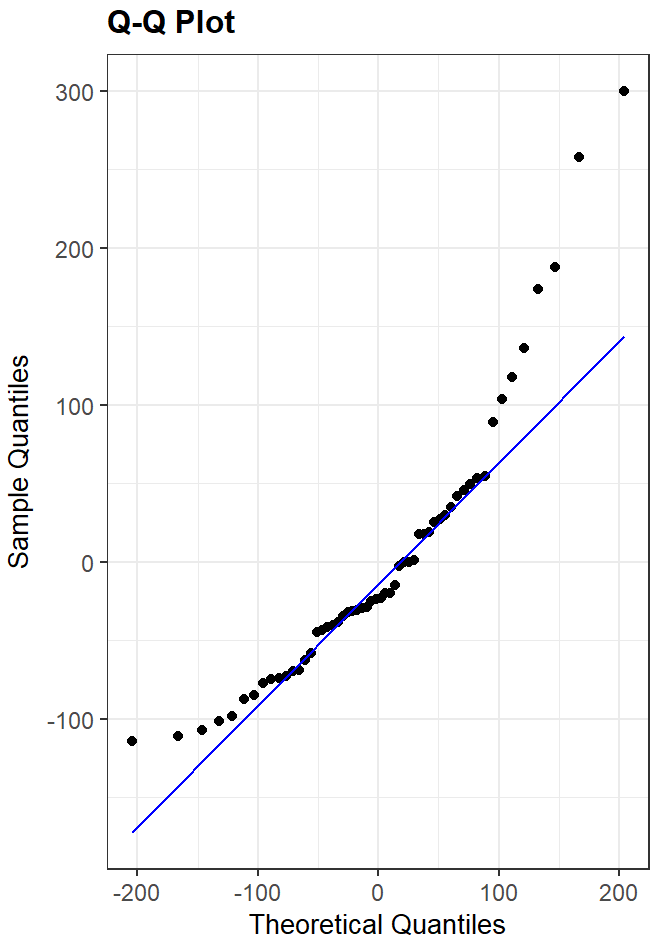
We should check the assumptions of our model to make sure that an MLR is the appropriate model choice. To do this we will use the `resid_panel()` function from last week.

```
library(ggResidpanel)
```

```
## Warning: package 'ggResidpanel' was built under R version 4.3.3
```

```
resid_panel(mod1, plots=c("resid","qq"))
```





The 4 conditions that we check for an MLR are the same as what we checked for an SLR. That is the residuals are independent and $\epsilon \sim N(0, \sigma^2)$.

Use the diagnostic plots to **check that the conditions for this model have been satisfied.**

▶ **CHECK**

Ordinarily at this point, you would try to fit a different model that doesn't have these violations in the assumptions of the residuals. Perhaps a model with an interaction (next week) or a polynomial (week 4) model would account for the non-normal residuals and the obvious bend in the residual scatter.

In general, we can say that `Length` and `Width` does influence perch `Weight`, since the trends are so strong between these variables. However, the estimates of the coefficients of `Length` and `Width` are probably not as reliable, since we were fitting straight lines to a relationship that was curvy. We wouldn't want to be making specific numerical inference using those β estimates from the model or using the model for prediction.

For now, let us put this problem aside and use this model to practice interpretations and predictions from a main effects multiple linear regression model. We will revisit some ways to deal with this issue in the coming weeks.

1.5 Use the Multiple Linear Regression model to make predictions.

We can use the same `predict()` command as last week to make predictions with our model. However, because we have multiple predictors in our model, we need to make sure that our `predict()` code includes a value for all predictors in the model.

```
predict(mod1, new=data.frame(Width=5, Length=30), interval="confidence")
```

```
##       fit      lwr      upr
## 1 417.962 391.7151 444.2089
```

With 95% confidence, when `Width` of perch is 5 cm and `Length` is 30 cm, mean `Weight` will be between 391.72 and 444.21 g.

```
predict(mod1, new=data.frame(Width=5, Length=30), interval="prediction")
```

```
##       fit      lwr      upr
## 1 417.962 238.1743 597.7497
```

With 95% confidence, when `Width` of one perch is 5 cm and `Length` is 30 cm, weight of the fish will be between 238.17 and 597.75 g.

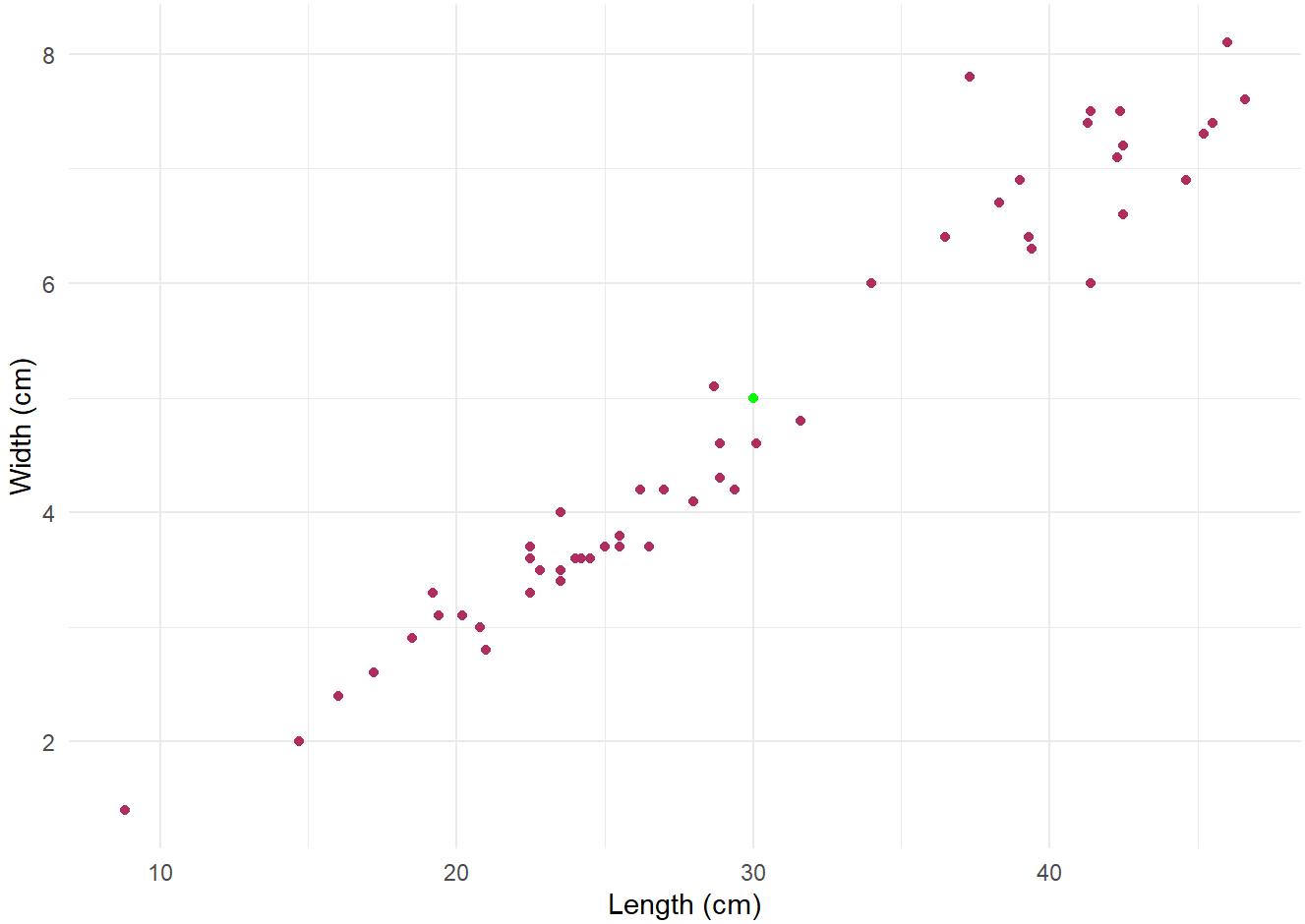
When making these predictions, the reliability can be influenced by three points:

1. **The r^2 value:** Larger r^2 values indicate that your model explains more the variation in your response. Low r^2 values won't impact the reliability of your intervals (you will end up with wider intervals generally though), but may impact the reliability of the fitted value for your predictions.
2. **The conditions of the residuals:** If conditions of the residuals are not met, then your predictions are going to be more unreliable.
3. **If you are extrapolating:** Predictions using extrapolation are often less reliable as you only have an idea of the true relationship through the data you have collected. Making predictions inside your range of observations (interpolation) is more reliable than those made outside of the observations (extrapolation).

You should check whether your predicted value is going to be made within your range of observations. One way to do this is to create a scatter plot of your 2 explanatory variables and see where the new point will fall in relation to your observations. Here we have added an extra point using `geom_point` and the x and y coordinates from the new point we want to predict.

NOTE: Here x=30 is used in the last line because Length=30 in our prediction and we have placed Length on the x axis in our plot. If we were to place Length on the y axis instead, then we would need to say y=30 in the last line.

```
#add predicted point to predictor exploratory plot
ggplot(data=per.df, aes(x=Length, y=Width))+
  geom_point(col="maroon", pch=16)+
  labs(y="Width (cm)", x="Length (cm)")+
  theme_minimal()+
  geom_point(x=30, y=5, colour="green")
```



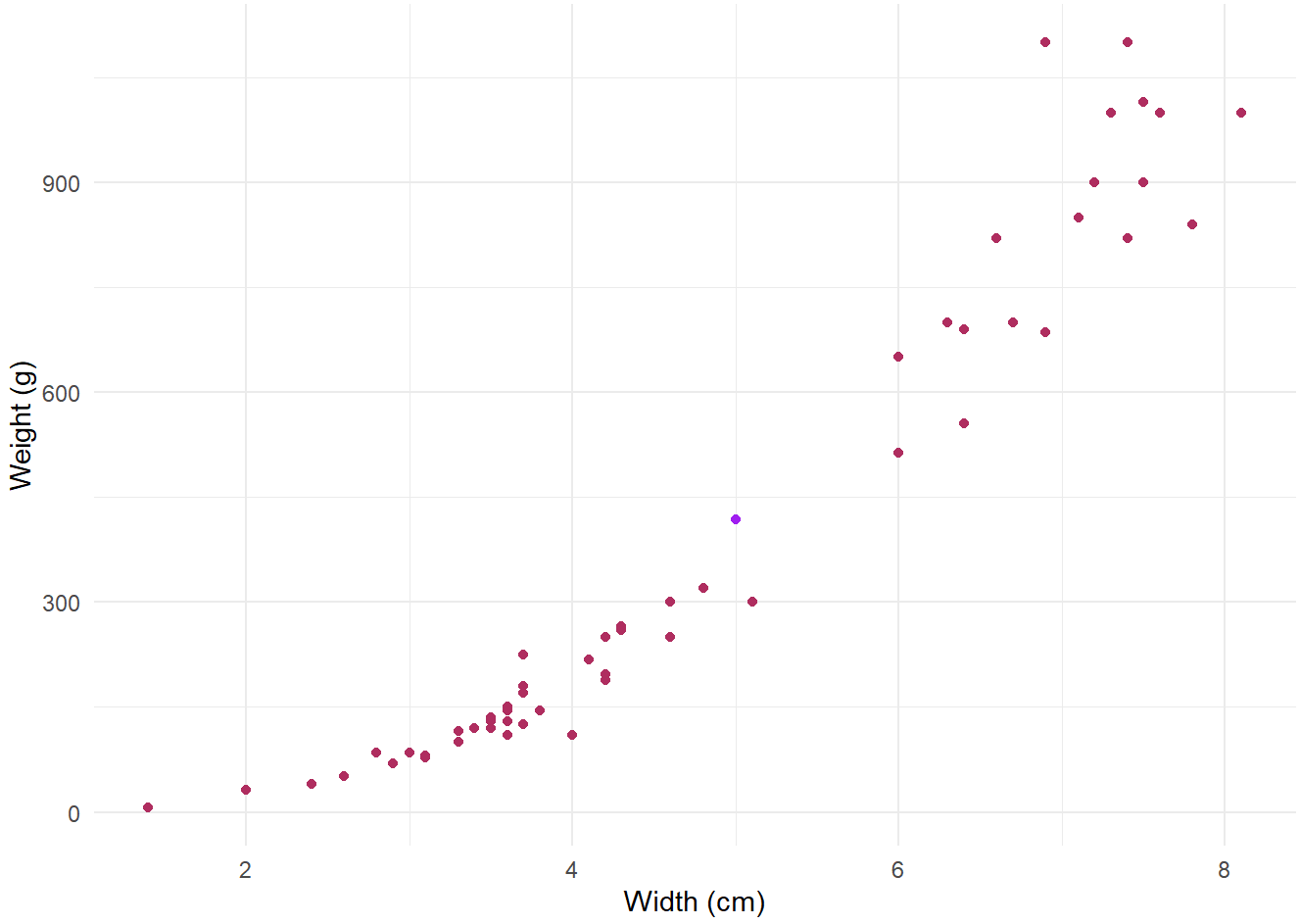
With this example, the point falls close to the scatter and is within the boundary of our observations, so this prediction is not extrapolation. However, the prediction is still unreliable as the assumed conditions for the residuals were not met for this model.

Make a plot to check whether a prediction for perch weight when length is 40cm and width is 4cm is extrapolation.

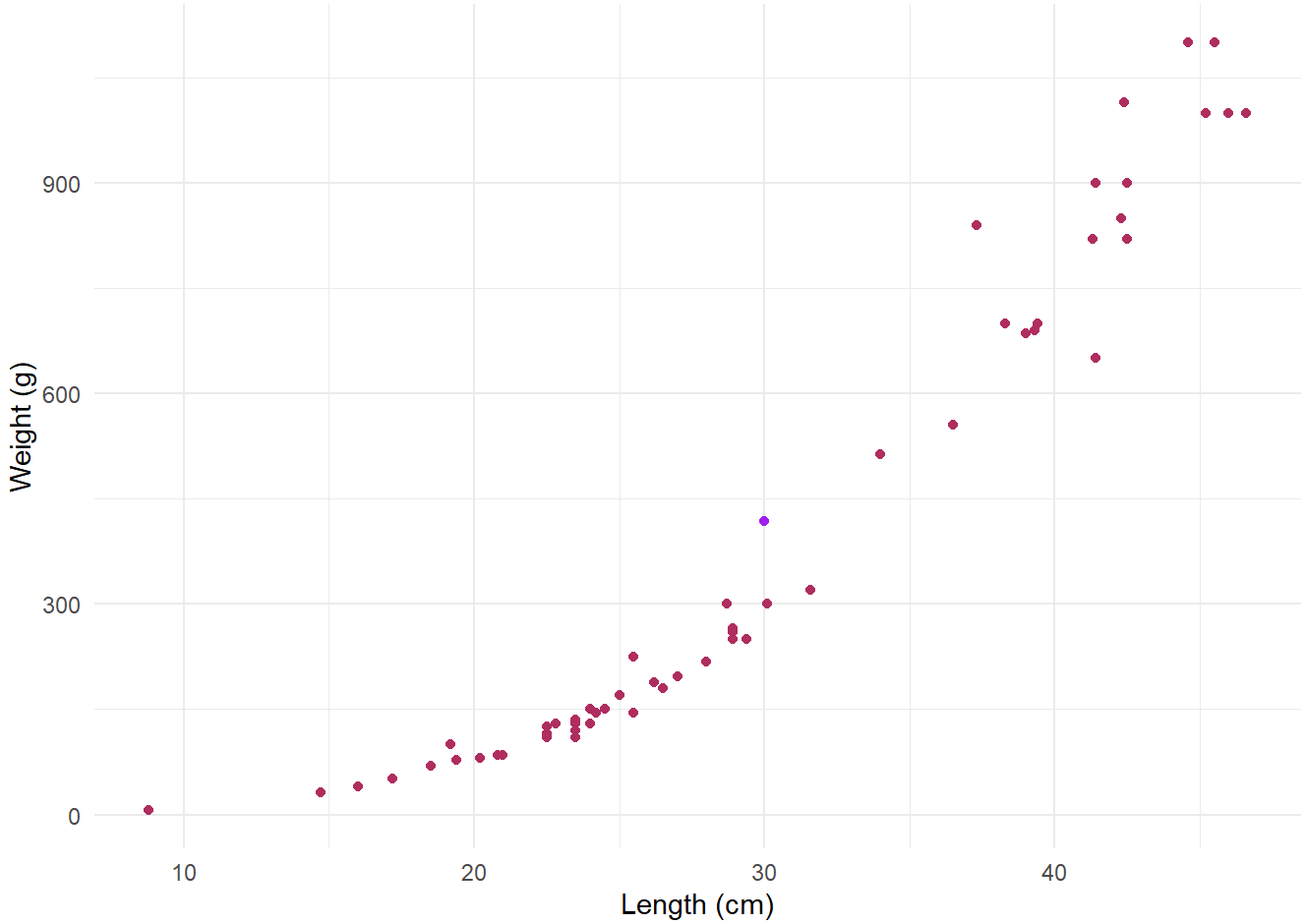
▶ **CHECK**

We mentioned earlier that the fitted values for our predictions are probably not reliable due to the violations in the conditions of the residuals. Lets see where our predicted point falls in relation to the scatter of our observations. To look at this, we will create a plot looking at the relationship between `Weight` and the predictors and add our predicted point.

```
#add predicted point to exploratory plot
ggplot(data=per.df, aes(x=Width, y=Weight))+
  geom_point(col="maroon", pch=16)+
  labs(y="Weight (g)", x="Width (cm)")+
  theme_minimal()+
  geom_point(x=5, y=417.96, colour="purple")
```




```
#add predicted point to exploratory plot
ggplot(data=per.df, aes(x=Length, y=Weight))+
  geom_point(col="maroon", pch=16)+
  labs(y="Weight (g)", x="Length (cm)")+
  theme_minimal()+
  geom_point(x=30, y=417.96, colour="purple")
```



We can see that our predicted point is in the ball park, but doesn't quite follow the observed trend. This is the main problem with fitting straight lines to relationships that are curved.

1.6 Write an informative conclusion

Finally, we should **write a conclusion detailing our main findings from our output.**

So far we have only looked at the coefficients from our model to define our model equation. Remember for an MLR we should also look at:

- The global utility p-value to assess the overall usefulness of the model.
- The p-value for each of our predictors; b_1 and b_2 to see if `Width` and `Length` are significant predictors of `Weight`.
- The r^2 to see how much of the variability in `Weight` is explained using a model with `Width` and `Length` as the predictors and,
- The confidence intervals for b_1 and b_2 to give us a range of where the true value of β_1 and β_2 might lie.

From the output we generated earlier:

```
## Analysis of Variance Table
##
## Response: Weight
##           Df  Sum Sq Mean Sq  F value  Pr(>F)
## Width      1 6179065 6179065 785.7974 < 2e-16 ***
## Length     1   50267   50267   6.3925 0.01447 *
## Residuals 53  416762    7863
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Call:
## lm(formula = Weight ~ Width + Length, data = per.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -113.86  -59.02  -23.29   30.93  299.85
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -578.758     43.667 -13.254  < 2e-16 ***
## Width        113.500     30.265   3.750 0.000439 ***
## Length        14.307      5.659   2.528 0.014475 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 88.68 on 53 degrees of freedom
## Multiple R-squared:  0.9373, Adjusted R-squared:  0.9349
## F-statistic: 396.1 on 2 and 53 DF,  p-value: < 2.2e-16
```

```
##                   2.5 %     97.5 %
## (Intercept) -666.343179 -491.17237
## Width         52.796272  174.20304
## Length         2.957273   25.65749
```

The global p-value for the model ($p < 2X10^{-16}$) is much less than 0.05, so a model containing `Length` and `Width` is a useful predictor of `Weight` in perch.

The p-value ($p = 0.0004$) for `Width` is much less than 0.05, so `Width` is a useful predictor of perch `Weight` when `Length` is also in the model.

The p-value ($p = 0.01$) for `Length` is less than 0.05, so `Length` is a useful predictor of perch `Weight` when `Width` is also in the model.

The adjusted r^2 value is 0.9349, indicating that 93.5% of the variability in perch `Weight` is explained by a model including `Length` and `Width`.

With 95% confidence, when `Length` of perch increase by 1 cm, the weight of perch increases between 2.96 and 25.66 g on average, when `Width` is kept constant.

With 95% confidence, when `Width` of perch increase by 1 cm, the weight of perch increases between 52.80 and 174.20 g on average, when `Length` is kept constant.

These conclusions should be made with caution as not all conditions of a linear regression where met for this model.

2 Practice Example: Timber Volume of Black Cherry Trees

Let's revisit the `trees` data set from last week.

The `trees` dataset in R contains measurements of the diameter (`Girth` in inches), `Height` (ft) and timber `Volume` (cubic ft) of timber in 31 felled black cherry trees.

Source: Forest Mensuration. H. A. Meyer (1953). Penns Valley Publishers, Inc.

Last week you were asked to model `Volume` using `Girth` or `Height` . This week your task is to **fit a multiple linear regression to predict the timber `Volume` of cherry trees using `Girth` and `Height` as explanatory variables.**

As part of fitting your model:

- Do some exploratory analysis using `ggpairs()` .
- Fit an MLR using the `lm()` command and write the model equation.
- Check assumptions of your model using the `resid_panel()` command.
- Write a meaningful conclusion using the output from the `anova()` , `summary()` and `confint()` commands.
- Predict mean timber `Volume` when `Height` of the cherry trees are 70 ft and when `Girth` of the cherry trees are 14 inches.
- Predict individual timber `Volume` of a tree when `Height` is 65 ft and `Girth` is 18 inces.

▶ **CHECK**

3 You're Finished!


MARINE BIOLOGY IS SO FASCINATING.
MICROBIOLOGY IS SO FASCINATING.
HIMELBLAU