Bayesian Statistics: Techniques and Models
Data Analysis Project: Concrete Strength

May 2023

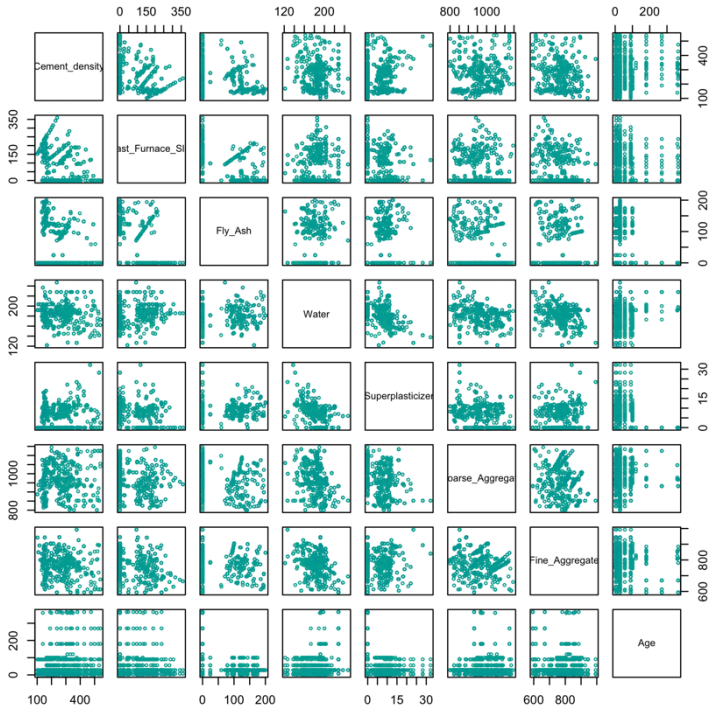
1. Introduction

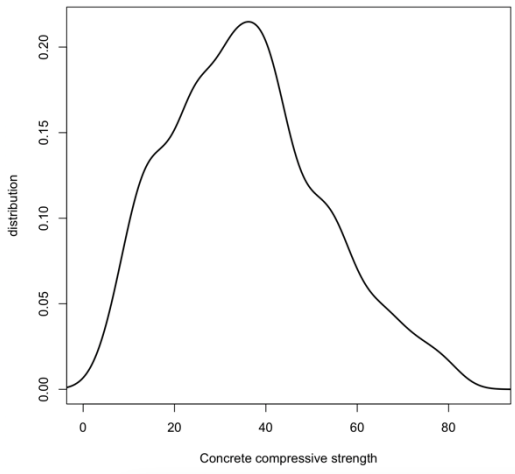
For this project, I chose a data set for concrete compressive strength which you can download from[1]. The concrete has different strength related with its composition and the way it was produced. Using the features from this dataset it should be possible model the expected strength (and use it to new cases).
With this project I intent to predict the concrete strength based on these features with a Bayesian linear model using MCMC with jags.

2. Data Exploration

The dataset has 8 features and a nineth feature being the concrete compressive strength which we want to model. The feature to be modelled are cement density, blast furnace slag, fly ash, water amount, duperplasticizer, coarse aggregate, fine aggregate all in kg/m^3 and Age in days. The concrete compressive strength (column 9) is in MPa.

Cement_density		Blast_Furnace_Slag		Fly_Ash		Water		Superplasticizer	
Min.	102.0	Min.	0.0	Min.	0.0	Min.	121.8	Min.	0.0
1st_Qu.	192.4	1st_Qu.	0.0	1st_Qu.	0.0	1st_Qu.	164.9	1st_Qu.	0.0
Median	272.9	Median	22.0	Median	0.0	Median	185.0	Median	6.4
Mean	281.2	Mean	73.9	Mean	54.2	Mean	181.6	Mean	6.2
3rd_Qu.	350.0	3rd_Qu.	142.9	3rd_Qu.	118.3	3rd_Qu.	192.0	3rd_Qu.	10.2
Max.	540.0	Max.	359.4	Max.	200.1	Max.	247.0	Max.	32.2
Coarse_Aggregate		**Age**		**Fine_Aggregate**		**Concrete compressive strength**			
Min.	801.0	Min.	594.0	Min.	1.0	Min.	2.3		
1st_Qu.	932.0	1st_Qu.	731.0	1st_Qu.	7.0	1st_Qu.	23.7		
Median	968.0	Median	779.5	Median	28.0	Median	34.5		
Mean	972.9	Mean	773.6	Mean	45.7	Mean	35.8		
3rd_Qu.	1029.4	3rd_Qu.	824.0	3rd_Qu.	56.0	3rd_Qu.	46.1		
Max.	1145.0	Max.	992.6	Max.	365.0	Max.	82.6		





In the table we can see the quantiles of the 9 features. In the bottom left set of plot we cannot see any eye evidence that the features are correlated between each other's which is good for our model, otherwise we might have needed to do some feature engineering. In

[1] https://archive.ics.uci.edu/dataset/165/concrete+compressive+strength

the right-hand side, we can see the distribution of the concrete strength which we want to predict.
The data should also be cleaned for large outlier or nulls/nans in the data. In our case the data is clean so we can use it directly.

3. Modelling

3.1. Postulate a model

For this case we will postulate a linear regression model, more complicated models are possible, but we will stick to this one. We will fit a Bayesian linear model and see which predictors affect the concrete strength. We will use jags and the model considered is:

```
mod_string = " model {
    for (i in 1:length(y)) {
        y[i] ~ dnorm(mu[i], prec)
        mu[i] = b[1] + b[2]*Cement_density[i] + b[3]*Blast_Furnace_Slag[i]
              + b[4]*Fly_Ash[i] + b[5]*Water[i] + b[6]*Superplasticizer[i]
              + b[7]*Coarse_Aggregate[i] + b[8]*Fine_Aggregate[i] + b[9]*Age[i]
        }

    for (i in 1:9) {
        b[i] ~ dnorm(0.0, 1.0/1.0e6)
    }

    prec ~ dgamma(5/2.0, 5*10.0/2.0)
    sig = sqrt( 1.0 / prec )
} "
```

```
mod = jags.model(textConnection(mod_string), data=data,
 inits=inits, n.chains=3)

Compiling model graph
  Resolving undeclared variables
  Allocating nodes
Graph information:
  Observed stochastic nodes: 1030
  Unobserved stochastic nodes: 10
  Total graph size: 11809

Initializing model
```

We used normal priors for the coefficients and inverse gamma prior for the variance.

3.2. Fit and check the model

Fit the model with normal likelihood and the previous priors with 10000 burn-in iterations in each 3 chains.

```
autocorr.diag(mod_sim)

             b[1]          b[2]          b[3]          b[4]          b[5]          b[6]          b[7]
Lag 0  1.0000000  1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
Lag 1  0.9998604  0.9846358 0.9252174 0.9100473 0.9982283 0.8423656 0.9988586
Lag 5  0.9992937  0.9444288 0.8695633 0.8298923 0.9923664 0.6531333 0.9944983
Lag 10 0.9985449  0.9091834 0.8486030 0.7887860 0.9850571 0.5917237 0.9893251
Lag 50 0.9918152 0.7775058  0.7691753 0.6454707 0.9304816 0.3629229 0.9541240
             b[8]          b[9]          sig
Lag 0  1.0000000 1.00000000 1.000000000
Lag 1  0.9985795 0.41463855 0.012378119
Lag 5  0.9930580 0.07129951 0.004986756
Lag 10 0.9865096 0.05991559 0.004496226
Lag 50 0.9437868 0.04186021 0.003711383
```

```
effectiveSize(mod_sim)

      b[1]       b[2]
  24.12475  511.44297
      b[3]       b[4]
 390.83739  826.85627
      b[5]       b[6]
 218.06943 2772.44345
      b[7]       b[8]
 147.07839  163.01744
      b[9]        sig
30738.70308 255568.71015
```

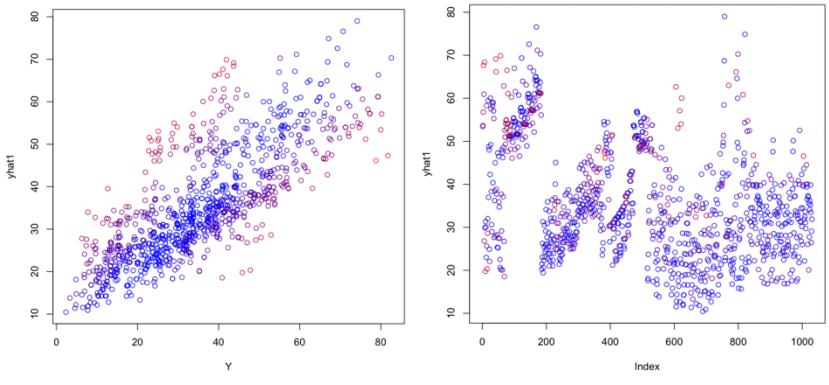
```
Mean of the coefficients:

          Mean       SD  Naive
b[1] -27.62768 28.137954
b[2]   0.12096  0.008773
b[3]   0.10524  0.010508
b[4]   0.08945  0.012962
b[5]  -0.14393  0.042410
b[6]   0.29815  0.094826
b[7]   0.01959  0.009945
b[8]   0.02174  0.011159
b[9]   0.11425  0.005437
sig   10.38536  0.229288
```

We can see that not all coefficients have fully converged, maybe there are not the better for this regression ad we should think about changing the model or remove some features. The age and Superplasticizer are the two most relevant feature for our model.

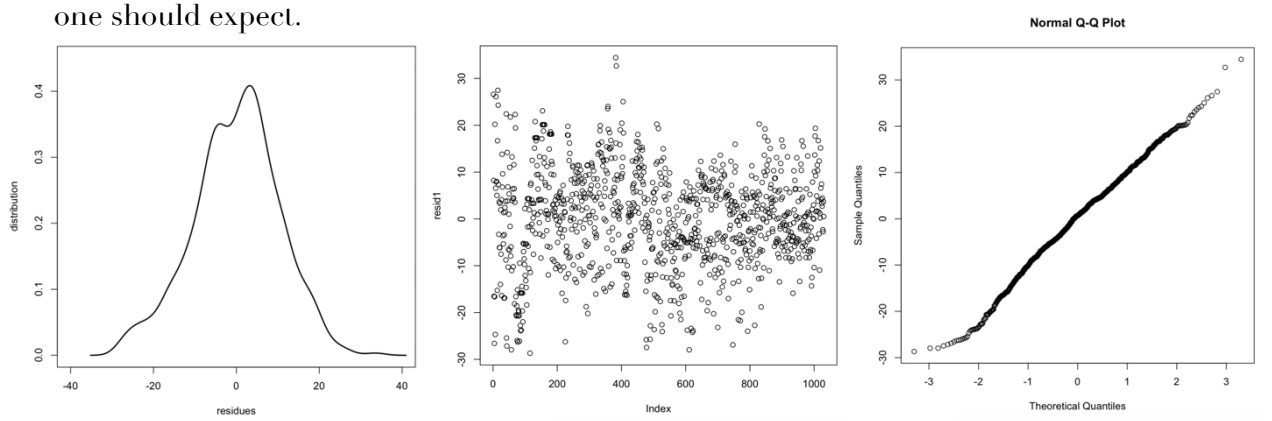
4. Results

4.1. Check the model



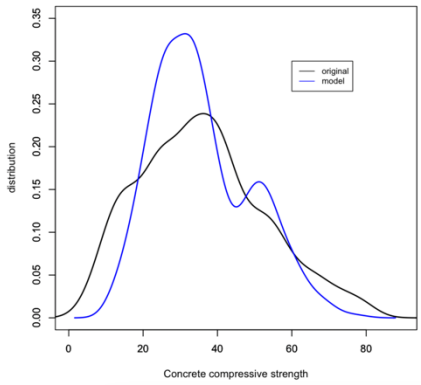
Using the model, we can predict the concrete strength with the trained feature (yhat) to study the models results.
In the left we can see the predicted strength in function of the actual value. We can see that we can predict the value but with a big variance. The colour if proportional to the residues. In the bottom left figure, we can also see that the yhat doesn't have any correlation with index.

Regarding the residues itself, you see the the next figures that the residue is mostly normal distributed as would be expected from our model. The residues of the predicted concrete strength along the index can be seen, and it doesn't have any dependency on the index as one should expect.



In the top right, we can see an almost perfect quantile distribution or the residues of the model.

4.2. Check and discussion



The distribution of the predicted strengths is quite similar to the real one. This shows that our model is quite good to have a approximated behaviour with low bias but have a very high variance with very high residues.
I believe the we should look at the feature with less significance and decide if they should be included, or do some feature engineering with them to improve the model.

4.3. Use the model

Here I calculated the probability that concrete with some specific days in age have a strength above 35 in kg/m^3. The strength 35.5 is the average strength in the sample and the Age seem the be the more relevant feature given its effective size. So, the results are for some points:

$$P(strengh > 35.5 \mid age = 30d) = 0.0019$$
$$P(strengh > 35.5 \mid age = 45d) = 0.77$$
$$P(strengh > 35.5 \mid age = 50d) = 0.9934$$