CHRISTOPHER LINDEMAN

MISSING DATA: THE MISSING LINK

Contents

List of Figures

Introduction

The field of data analytics is a massive one, and every tool of the data professional relies on one resource: good data. Notice that I said "good data" and not just "data". Data is useful only as much as we know it to be true, or at least an accurate representation of the truth. One could argue that the primary job of a skilled data professional is to ensure the quality of the data used to present or model, then proceed to present or model it. Thus it is imperative that we have good data to use, and when we lack the data we need, we have a reasonable way of estimating it, which is the inspiration for this writing.

Missing Links

Missing Data is common. As computing becomes more ubiquitous, the amount of data being generated is growing exponentially, and along with it the amount of missing data is also growing. Doing good work with data means understanding this missing data. Sometimes missing data is pretty random - a sensor fails to record the current temperature because a battery dies. Sometimes missing data is a bit less random, like when I was younger and would constantly forget to clock in at my grocery store job. It was random, but no one else seemed to have that problem. Then we have an almost predictable kind of randomness, like how sick people are more likely to miss work, or how families in affluent neighborhoods tend to pay lower interest rates. These are kind of situations that are features of the systems we're operating in, but certainly mislead our predictions when the data are missing or poorly estimated.

What about omission? It's reasonable to ask "why not simply omit that data?". Well, there are a few reasons that omission might be detrimental. First, it's common knowledge that more data makes better models, right? Well, that's true if the data is good data. But more importantly, what if the datum that we omit has some important influence? I'm not talking about the effect of Elon Musk's wealth on the distribution of American incomes. This is more about the intersection of where the data fits in addition to its value, like how we may have missing data due to underreporting on surveys for sensitive behaviors the skews our belief in how bad a problem is (e.g. drug abuse being underreported allows us to assume it is less prevalent than it is).

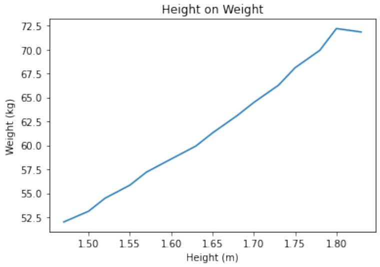
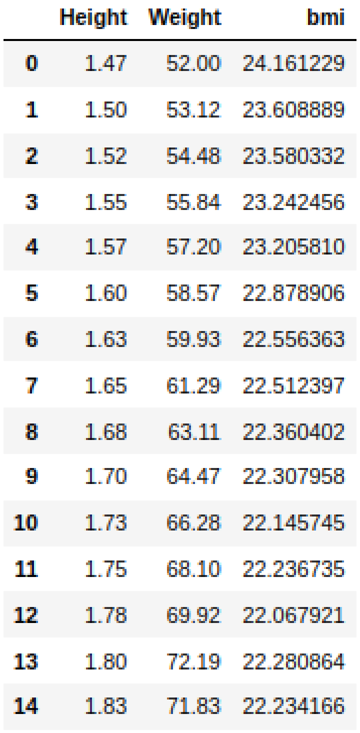


Figure 1: Shows linearity of the dataset.



	Height	Weight	bmi
0	1.47	52.00	24.161229
1	1.50	53.12	23.608889
2	1.52	54.48	23.580332
3	1.55	55.84	23.242456
4	1.57	57.20	23.205810
5	1.60	58.57	22.878906
6	1.63	59.93	22.556363
7	1.65	61.29	22.512397
8	1.68	63.11	22.360402
9	1.70	64.47	22.307958
10	1.73	66.28	22.145745
11	1.75	68.10	22.236735
12	1.78	69.92	22.067921
13	1.80	72.19	22.280864
14	1.83	71.83	22.234166

Figure 2: This data is used for all analysis of popular methods.

Example Data

For the sake of simplicity, I've created a small data set of heights and weights. In actuality, the subject of this data is irrelevant. What matters is that it is a toy data set with a linear dependence. The small sample size and highly correlated data will help illustrate the effects of different methods. For illustration purposes, I've made a point of maintaining the same axis scale for plots to more easily visualize their differences. Since BMI is a direct function of height and weight, it will not be used for analysis. It is part of the data simply for reference.

Types of Missingness

As we've seen, there are different types of missing data, and each has its own important implications when it comes to the task of data wrangling.

Missing Completely at Random (MCAR)

There is data that is missing completely at random. When a data entry clerk mistypes something, that's MCAR. When a battery dies and some piece of datum isn't recorded, that's MCAR. When a file gets corrupted and some of the data are compromised, that's MCAR.

Missing at Random (MAR)

Missing at Random is a bit of a misnomer, because we're assuming that there is a reason for the missingness, but that reason isn't explained by the data. If the 46 bus is late every morning after the driver goes out to celebrate his favorite baseball team's win, it's pretty likely that it will seem random, although it certainly is because Jeff lacks self control when the Pirates win. So, it's not random. It's really predictable, but to an outsider, it sure seems random.

Missing Not at Random (MNAR)

This can be thought of as missingness due to specific underlying factors. I like to think of this as structured missingness. Surveys are rife with this. People with high incomes often omit reporting it on surveys. People who are chronically ill tend to miss work more than healthy people. People in poor neighborhoods tend to pay higher interest rates than those in affluent areas. MNAR is similar to MAR, except that in MNAR cases, we can generalize a bit more, and the outlier of Jeff making poor decisions is more akin to trends the we can understand from the data if we look for it.

Effects and Importance of Missing Data

MCAR

It seems like missingness of this sort exists solely for the reason of irritating data professionals. Luckily, it's not such a big deal. Because there isn't some underlying reason for the data to be missing, removing it has no effect on the bias of the data, though it may affect the variance, particularly if the data set is small.

MAR

As for the data itself, MAR data is a mystery, and may require further investigation. MAR is usually not grouped, and while it may have underlying meaning, it may not make a major impact on any reports or models created from this data. That is, unless your transit authority has a lot of Jeff's working for it. At some level of Jeff-iness, we'd consider it MNAR.

MNAR

This is undoubtedly the most important class of missing data. There is definite structure to this data, and it has implications greater than those of some outlier like our buddy Jeff the bus driver. These are effects that exist for a reason, and are widespread. If a traffic sensor never works at morning rush hour, that is very important. We can model them, but we need to be careful. MNAR will be the focus of our investigation as it is interesting and important.

Determining the Type of Missingness

As we've established, understanding the type of missingness in the data is essential to understanding the impact and, hence, how we can handle these missing values. In determining the difference between MCAR and MAR, we can use Little's test for difference. This differentiates the less difficult cases.

This brings us to differentiating between MAR and MNAR, which are much more important. Since the only way to truly know if something is MNAR or MAR is to measure the missing data (a luxury we do not possess in the world of real data), we must either rely on our own expertise of the field in which we're performing analysis or assume that any data that is not MCAR be treated with the same level of care as we would treat MNAR data. Our investigation should point us to that technique.

Popular Methods

With the increasing availability of data and computing power, it is becoming common for amateurs to attempt complicated analysis projects in an effort to simplify their lives, make better decisions, or develop a new product. An unfortunate by-product of the ease with which beginners can do work with data is the plethora of bad data practices and popular advice that is proliferating itself across blogs, tutorials, and instructional videos. Participating in a data science bootcamp inspired me to pursue a graduate degree in analytics, and I'm still working to unlearn several naïve practices I picked up from that bootcamp.

Omission

Here we're talking about omitting the entire observation, or if this predictor has many missing variables, removing the predictor, from the dataset.

In cases of MCAR, this tends to have minimal impact on the usefulness of the resulting model, as truly random removal will not affect the bias of the model. However, the variance of the model will be affected, particularly by reducing the degrees of freedom, which is the cause of the greater uncertainty associated with small sample sizes.

This effect is intensified as more data more from the extremes are removed. Perhaps underweight or overweight respondents didn't feel comfortable reporting their weight? This certainly shifts the reported mean, creating a new baseline for average.

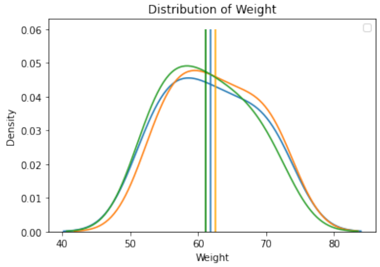


Figure 3: An example a MAR. The least or most heavy did not report. This makes a small shift in the mean of the data and a slight distortion of the distribution.

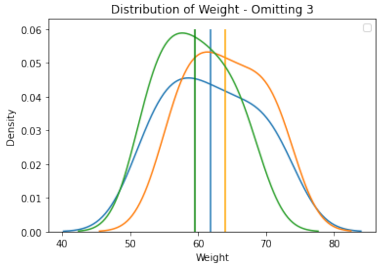


Figure 4: The more important MNAR case, where smaller and larger women are not reporting. We can see that the means shifts significantly and the kurtosis of the distribution is growing, along with a greater distortion of the distributions.

Donor-Based Imputation

Approximating the missing data from the existing data (donor) is a common tool that is fairly easy to implement. However, the effects of these imputation methods may have unwanted side-effects, which are enhanced by the proportion of the missing values to the sample size. This is where we begin to see strange things happening.

Mean

The simplest of imputation methods, it's also probably the most commonly applied. It's the one I was taught as a default in my "highly respected" data science bootcamp. Mean imputation substitutes the mean value of a predictor for the missing value. This leads to a model with smaller variance than should be expected, as the data are now more centered about the mean. This form of imputation may also shift the mean of the data, particularly when the missing observation is an edge case.

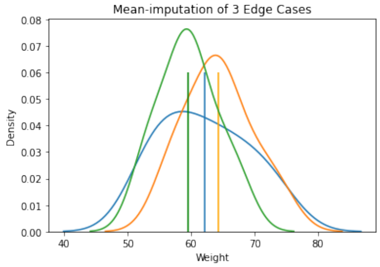


Figure 5: Distributions of the data using mean-imputation of missing data for the 3 least and 3 greatest weights.

Hot-deck

Hot-deck is a very old method, that works fast, but you can't ever really tell what you're doing to the data, because the order of the observations affects the value that will be imputed. For an observation i and predictor j, $X_{i,j}$, if $X_{i+1,j}$ is missing, we simply let $X_{i+1,j} = X_{i,j}$.

This is likely to present less of a problem for time series data, as the basis for time series is that $S_{t+1} = S_t + X$ for some X.

The effect of this form of imputation is highly dependent on where the missing datum lie and if there are several missing in a chunk.

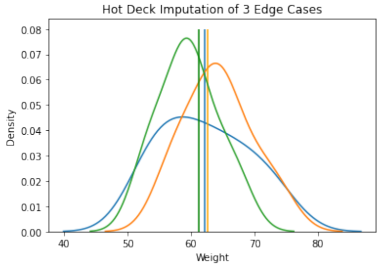


Figure 6: Distributions of the data using hot deck imputation of missing data for the 3 least and 3 greatest weights.

Comparing Popular Methods

In each MNAR case, popular methods cause the kurtosis to grow, except in the case of the central values being missing for the hot deck imputation. However, we notice that the distribution itself shifts from somewhat normal to a more bivariate one.

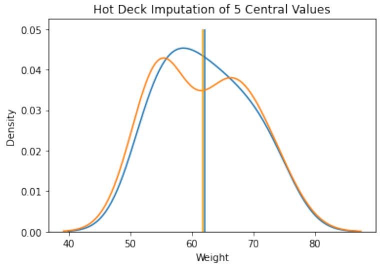


Figure 7: Distributions of the data using hot deck imputation of missing data for the 3 least and 3 greatest weights.

Estimation from GIBBS Sampling

GIBBS Sampling is a MCMC method that draws an instance from the distribution of each variable conditional on the other variables, in order to estimate complex joint distributions. Those last few words are important 'complex joint distributions'. That says we're looking at the effects of more than one variable. If we were to sample from a distribution whose mean was equal to the mean of the predictor whose value is missing, we'd end up doing a lot of computing to end up with roughly the same answer as simple mean-imputing. GIBBS is effective for multiple linear regression, so let's have a look at a more complex dataset for this.

Further, GIBBS sampling requires us to make some assumptions about the underlying structure of the data. A linear regression is modelled as:

$$y = X\beta + \epsilon$$
$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$

So we have two undefined variables that we need to somehow build a prior distribution from which to sample. We can let $\beta \sim \mathcal{N}(\mu, \tau^2)$ where μ is our best approximation of the average value of β and τ^2 is an approximation of the variability of our estimate. A good choice for the distribution of σ^2 is an inverse gamma with parameters a, b. In practice, an uninformative prior (small parameters) on τ may be very effective.

In the Bayesian manner, we have:

$$p(\beta, \sigma^2 | y) \propto p(y|\beta, \sigma^2)p(\beta)p(\sigma^2)$$

Luckily WinBUGS has a pretty simple interface for this.

For each predictor, we set up an equation for the model to generate an estimate.

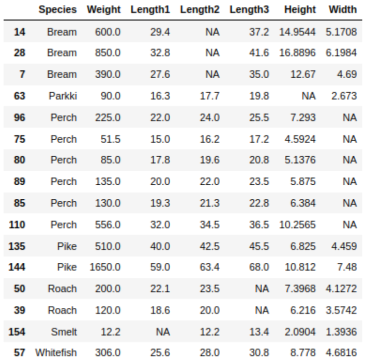
$$X_1 = X_2\beta_1 + X_3\beta_2 + X_4\beta_3 + X_5\beta_4 - Y\beta_5$$

We use this schema for all predictors to generate an estimate of the missing values. Running the sampler for 100,000 iterations, we've arrived at a stable and solid estimate for all predictor variables.

Data

Comparative analysis is to be performed on the fish market dataset from Kaggle. This is a linear regression problem with one categorical variable (Species), and 5 continuous predictors. In order to make this a MNAR case, I chose to remove all of one predictor for 5 of the 7 species. Pike and Whitefish have complete data. We can think of this as possible if there are some types of measurements that aren't done for particular species. Maybe it's tradition. In any case, it is certainly not at random, and that's important.

After generating the missing values in WinBUGS, the GIBBS-imputed dataframe is complete.



	Species	Weight	Length1	Length2	Length3	Height	Width
14	Bream	600.0	29.4	NA	37.2	14.9544	5.1708
28	Bream	850.0	32.8	NA	41.6	16.8896	6.1984
7	Bream	390.0	27.6	NA	35.0	12.67	4.69
63	Parkki	90.0	16.3	17.7	19.8	NA	2.673
96	Perch	225.0	22.0	24.0	25.5	7.293	NA
75	Perch	51.5	15.0	16.2	17.2	4.5924	NA
80	Perch	85.0	17.8	19.6	20.8	5.1376	NA
89	Perch	135.0	20.0	22.0	23.5	5.875	NA
85	Perch	130.0	19.3	21.3	22.8	6.384	NA
110	Perch	556.0	32.0	34.5	36.5	10.2565	NA
135	Pike	510.0	40.0	42.5	45.5	6.825	4.459
144	Pike	1650.0	59.0	63.4	68.0	10.812	7.48
50	Roach	200.0	22.1	23.5	NA	7.3968	4.1272
39	Roach	120.0	18.6	20.0	NA	6.216	3.5742
154	Smelt	12.2	NA	12.2	13.4	2.0904	1.3936
57	Whitefish	306.0	25.6	28.0	30.8	8.778	4.6816

Figure 8: Sample of the Fish Market dataset after replacing values MNAR.

Note For the comparison, we will not be considering hot-deck imputation. This is because we are replacing all values of a predictor, and we can simply intuit that randomly choosing a value (as the ordering here is random) would be an irresponsible application of randomness. Omission of data is also impractical as well, as only 22 rows now exist without missing data. Thus for data with unstructured ordering, or data with a lot of missing values, it is only reasonable to consider simple imputation (e.g. mean or median) or GIBBS-sampled imputation.

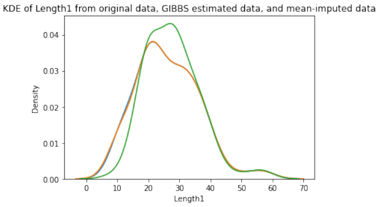


Figure 9: Length1 kernel density estimates (KDEs)

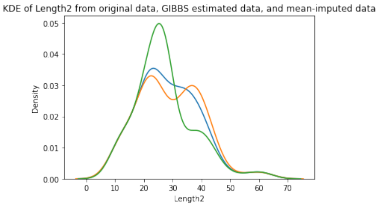


Figure 10: Length2 KDEs

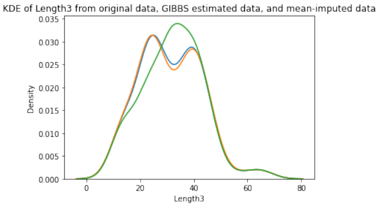


Figure 11: Length3 KDEs

Initial Comparison

Both the mean-imputed and the GIBBS-imputed data are prepared for a simple linear regression model in the sklearn package by first creating dummy variables for the factor 'Species'.

Also, we compute the total absolute difference between the imputed datasets and the base case. The linear regression is run on 100 random states with train/test sizes of 85/15,80/20,75/25,70/30.

Surprisingly, the mean-imputed model proves to have greater explanatory power for this data.

Impute Method	Absolute Difference	R-squared	BIC
GIBBS	228.51	.931	594.18
mean	691.27	.877	580.97

WHOA! Could this be some fluke? Intuitively, better data creates a better model, and a model with better explanatory power is better. How can we have better model if the data is not as good?

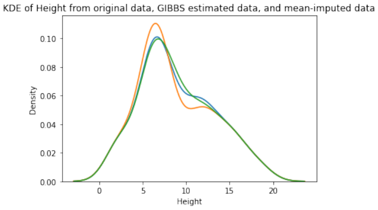


Figure 12: Height KDEs

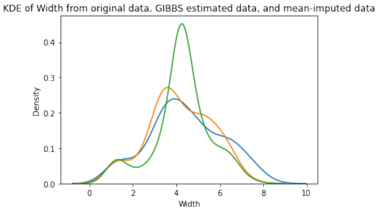


Figure 13: Width KDEs

Selection Criteria

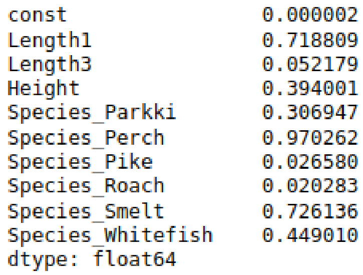
First, let us recall that the accuracy score of a model does not mean that it is actually a good fit for the data. We can perform a goodness of fit test to see if the modeling scores are indicative of the quality of the imputation method. The figures on the right very clearly show that imputing from GIBBS sampling seem to follow the distribution of the true data much more accurately. To be precise, we perform the Kolmorogov-Smirnov test with the following results:

Kolmogorov-Smirnov Test Results

Predictor	GIBBS statistic	GIBBS p-value	mean statistic	mean p-value
Length1	0.031	.999	.088	.570
Length2	.088	.57	.201	.003
Length3	.019	.999	.119	.207
Height	.057	.962	.038	.999
Width	.107	.324	.195	.005

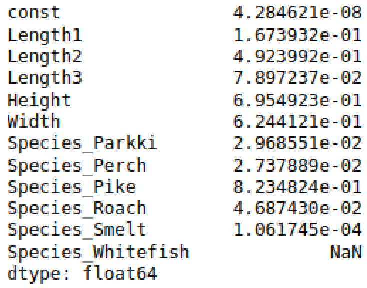
Kolmogorov-Smirnov is a hypothesis test with H_0 : the distributions of the data are the same. Since the p-vales of the GIBBS-imputed data are higher than any reasonable critical point, we cannot reject H_0 and assume that the distributions are the same. For the mean-imputed data, we must reject H_0 for Length2 and for Width, and conclude that the distributions are not the same. This is apparent in the figures. This tells us that mean-imputation for the variables with rejected hypotheses are not a good fit, and would generally be left out of a model that we'd consider to be a good one. Thus we have these models for comparison:

- $\hat{y} = \beta_0 + \beta_1 Length1 + \beta_2 Length2 + \beta_3 Length3 + \beta_4 Height + \beta_5 Width$ using the GIBBS-imputed data
- $\hat{y} = \beta_0 + \beta_1 Length1 + \beta_3 Length3 + \beta_4 Height$ using the mean-imputed data



```
const                0.000002
Length1              0.718809
Length3              0.052179
Height               0.394001
Species_Parkki       0.306947
Species_Perch        0.970262
Species_Pike         0.026580
Species_Roach        0.020283
Species_Smelt        0.726136
Species_Whitefish    0.449010
dtype: float64
```

Figure 14: p-values of the reduced model with mean-imputed values



```
const                4.284621e-08
Length1              1.673932e-01
Length2              4.923992e-01
Length3              7.897237e-02
Height               6.954923e-01
Width                6.244121e-01
Species_Parkki       2.968551e-02
Species_Perch        2.737889e-02
Species_Pike         8.234824e-01
Species_Roach        4.687430e-02
Species_Smelt        1.061745e-04
Species_Whitefish             NaN
dtype: float64
```

Figure 15: p-values of the full model with GIBBS-imputed values

Final Comparison

Testing these models, we're finally beginning to make sense.

Model	R-squared	AIC	BIC	RMSE	SSR
full-GIBBS	.958	573.6	594.18	789.4	275105
reduced-mean	.955	575.7	594.39	842.7	299536

But wait! What about the adjusted R^2? After all, the model with mean-imputed data has fewer variables. The adjusted R^2 values are .9445 for the reduced mean model and .9463 for the full GIBBS model.

Perhaps the most important aspect of this analysis is that the only variables deemed significant at the 95% confidence for the mean-imputed model, are the constant and two of the Species dummy variables, of which one (Pike) has no imputed data. On the other hand, for the full-GIBBS model, all variables are significant at the 99% confidence level. The NaN p-values for $Species\_Whitefish$ indicates that a probability of exactly 0 or 1 on a logit scale has been returned. It is a feature of the computation. In this case, I believe it is safe to assume we have exactly 0 for this p-value.

It is important to note that higher R^2 always indicates a better model, however this is only the case when the model assumptions are fulfilled. We can revisit these assumptions as a way to justify our confidence that the goodness of fit is a better metric for comparison by having a look at the pairplot of the variables, with particular attention paid to to the first line, where we see that the assumptions of normality and homogeneity are clearly violated.

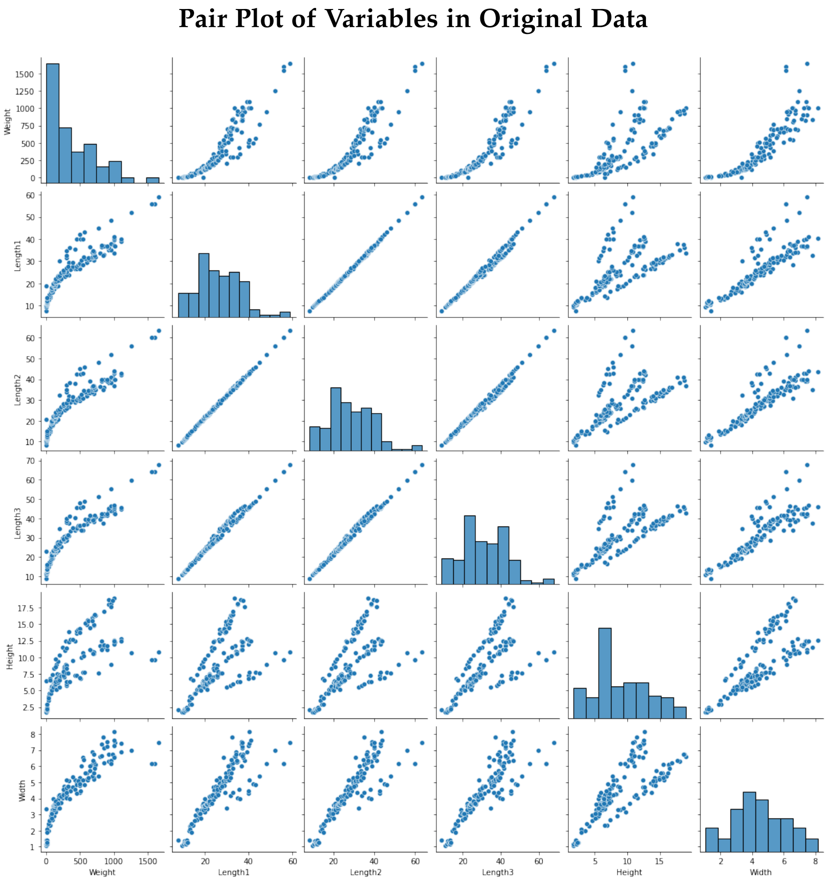


Figure 16: The relationship of the variables show violation of linear regression assumptions.

Conclusion

The shift in the distribution of data that occurs from all imputation methods affects the representation of the modified data. This can manifest itself as a smaller confidence interval and the consequence of increased kurtosis as in the case of mean imputation, with MNAR data being most effected, or simply by distorting the distribution and being a less-good fit for the data which occurs any time data is altered or fabricated. However, we see clearly that the data is more accurately represented when using imputation by GIBBS sampling. It may still be worth considering a simple imputation for very small amounts of missing data, which seem to have little affect. However, in cases where there is much missing data, or small missing data for several predictors, sampling should be the preferred method of data-imputation.

Afterword

The amount of attention to detail with which we undertake a data analysis task is proportional to the ease with which we complete it. Throughout this project, I had a good notion as to what to expect, and my eagerness to gather results led to a dead-end regarding the initial R^2 values for the GIBBS/mean-imputation comparison. Thanks to Aaron Redding, Harpreet Matharoo, and Aurimas Racas for their input, which likely saved many hours of confusion and forced me to revisit my own process to perform the analysis from first principles. It is easy to get flustered when we know what to expect, and the instinct to rush forward could have led to a very incorrect conclusion. I hope this inspires you to approach your analysis tasks with the utmost care and provides an impetus for changing how you impute missing values. I know it has improved how I approach missing data.

Appendix

Data

The data used can be found on my repository along with all figures found in this paper, as well as the BUGS code for imputing data. Height/weight data is the 'data.csv' file and 'Fish.csv' is the fish market data. 'fish_impute_l2.odc' is the BUGS code. My repository:

https://github.com/slimhindrance/ImputationAnalysis

Analysis Code

The file 'project analysis.ipynb' contains a Jupyter notebook that I used in performing all of the analysis for this project. It is currently a bit haphazard, and some cells may not run in order. I will be working to clear that up as time permits.

Complete BUGS Code

The following code was used to generate the missing data for Weight in the dataset. For each other predictor (l1,l2,l3,h) simply swap the predictor and w in the first 10 lines of the model code.

```
model {
for (i in 1:159) {
w[i] ~ dnorm(mu[i], tau)
mu[i] <- alpha + beta1*l2[i] -beta2*y[i] + beta3*l1[i] + beta4*l3[i] + beta5*h[i]
l3[i] ~ dnorm(0, 0.001)
l1[i] ~ dnorm(0, 0.001)
h[i] ~ dnorm(0, 0.001)
l2[i] ~ dnorm(0, 0.001)
}
alpha ~ dnorm(0, 0.001)
beta1 ~ dnorm(0, 0.001)
```

```
beta2 ~ dnorm(0, 0.001)
beta3 ~ dnorm(0, 0.001)
beta4 ~ dnorm(0, 0.001)
beta5 ~ dnorm(0, 0.001)
tau <- 1/sigma2
log(sigma2) <- 2*log.sigma
log.sigma ~ dflat()
}
Data:
list(y=c(242.0, 290.0, 340.0, 363.0, 430.0, 450.0, 500.0, 390.0, 450.0,
500.0, 475.0, 500.0, 500.0, 340.0, 600.0, 600.0, 700.0, 700.0, 610.0, 650.0,
575.0, 685.0, 620.0, 680.0, 700.0, 725.0, 720.0, 714.0, 850.0, 1000.0, 920.0,
955.0, 925.0, 975.0, 950.0, 40.0, 69.0, 78.0, 87.0, 120.0, 0.0, 110.0, 120.0,
150.0, 145.0, 160.0, 140.0, 160.0, 169.0, 161.0, 200.0, 180.0, 290.0, 272.0,
390.0, 270.0, 270.0, 306.0, 540.0, 800.0, 1000.0, 55.0, 60.0, 90.0, 120.0,
150.0, 140.0, 170.0, 145.0, 200.0, 273.0, 300.0, 5.9, 32.0, 40.0, 51.5, 70.0,
100.0, 78.0, 80.0, 85.0, 85.0, 110.0, 115.0, 125.0, 130.0, 120.0, 120.0,
130.0, 135.0, 110.0, 130.0, 150.0, 145.0, 150.0, 170.0, 225.0, 145.0, 188.0,
180.0, 197.0, 218.0, 300.0, 260.0, 265.0, 250.0, 250.0, 300.0, 320.0, 514.0,
556.0, 840.0, 685.0, 700.0, 700.0, 690.0, 900.0, 650.0, 820.0, 850.0, 900.0,
1015.0, 820.0, 1100.0, 1000.0, 1100.0, 1000.0, 1000.0, 200.0, 300.0, 300.0,
300.0, 430.0, 345.0, 456.0, 510.0, 540.0, 500.0, 567.0, 770.0, 950.0, 1250.0,
1600.0, 1550.0, 1650.0, 6.7, 7.5, 7.0, 9.7, 9.8, 8.7, 10.0, 9.9, 9.8, 12.2, 13.4,
12.2, 19.7, 19.9), l1=c(23.2, 24.0, 23.9, 26.3, 26.5, 26.8, 26.8, 27.6, 27.6,
28.5, 28.4, 28.7, 29.1, 29.5, 29.4, 29.4, 30.4, 30.4, 30.9, 31.0, 31.3, 31.4,
31.5, 31.8, 31.9, 31.8, 32.0, 32.7, 32.8, 33.5, 35.0, 35.0, 36.2, 37.4, 38.0,
12.9, 16.5, 17.5, 18.2, 18.6, 19.0, 19.1, 19.4, 20.4, 20.5, 20.5, 21.0, 21.1,
22.0, 22.0, 22.1, 23.6, 24.0, 25.0, 29.5, 23.6, 24.1, 25.6, 28.5, 33.7, 37.3,
13.5, 14.3, 16.3, 17.5, 18.4, 19.0, 19.0, 19.8, 21.2, 23.0, 24.0, 7.5, 12.5,
13.8, 15.0, 15.7, 16.2, 16.8, 17.2, 17.8, 18.2, 19.0, 19.0, 19.0, 19.3, 20.0,
20.0, 20.0, 20.0, 20.0, 20.5, 20.5, 20.7, 21.0, 21.5, 22.0, 22.0, 22.6, 23.0,
23.5, 25.0, 25.2, 25.4, 25.4, 25.4, 25.9, 26.9, 27.8, 30.5, 32.0, 32.5, 34.0,
34.0, 34.5, 34.6, 36.5, 36.5, 36.6, 36.9, 37.0, 37.0, 37.1, 39.0, 39.8, 40.1,
40.2, 41.1, 30.0, 31.7, 32.7, 34.8, 35.5, 36.0, 40.0, 40.0, 40.1, 42.0, 43.2,
44.8, 48.3, 52.0, 56.0, 56.0, 59.0, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA), l2=c(NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, 14.1, 18.2, 18.8, 19.8, 20.0, 20.5, 20.8, 21.0, 22.0, 22.0, 22.5, 22.5,
22.5, 24.0, 23.4, 23.5, 25.2, 26.0, 27.0, 31.7, 26.0, 26.5, 28.0, 31.0, 36.4,
40.0, 14.7, 15.5, 17.7, 19.0, 20.0, 20.7, 20.7, 21.5, 23.0, 25.0, 26.0, 8.4,
13.7, 15.0, 16.2, 17.4, 18.0, 18.7, 19.0, 19.6, 20.0, 21.0, 21.0, 21.0, 21.3,
22.0, 22.0, 22.0, 22.0, 22.0, 22.5, 22.5, 22.7, 23.0, 23.5, 24.0, 24.0, 24.6,
25.0, 25.6, 26.5, 27.3, 27.5, 27.5, 27.5, 28.0, 28.7, 30.0, 32.8, 34.5, 35.0,
36.5, 36.0, 37.0, 37.0, 39.0, 39.0, 39.0, 40.0, 40.0, 40.0, 40.0, 42.0, 43.0,
43.0, 43.5, 44.0, 32.3, 34.0, 35.0, 37.3, 38.0, 38.5, 42.5, 42.5, 43.0, 45.0,
46.0, 48.0, 51.7, 56.0, 60.0, 60.0, 63.4, 9.8, 10.5, 10.6, 11.0, 11.2, 11.3,
11.8, 11.8, 12.0, 12.2, 12.4, 13.0, 14.3, 15.0), l3=c(30.0, 31.2, 31.1, 33.5,
34.0, 34.7, 34.5, 35.0, 35.1, 36.2, 36.2, 36.2, 36.4, 37.3, 37.2, 37.2, 38.3,
38.5, 38.6, 38.7, 39.5, 39.2, 39.7, 40.6, 40.5, 40.9, 40.6, 41.5, 41.6, 42.6,
44.1, 44.0, 45.3, 45.9, 46.5, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 28.7, 29.3, 30.8,
34.0, 39.6, 43.5, 16.5, 17.4, 19.8, 21.3, 22.4, 23.2, 23.2, 24.1, 25.8, 28.0,
29.0, 8.8, 14.7, 16.0, 17.2, 18.5, 19.2, 19.4, 20.2, 20.8, 21.0, 22.5, 22.5,
22.5, 22.8, 23.5, 23.5, 23.5, 23.5, 23.5, 24.0, 24.0, 24.2, 24.5, 25.0, 25.5,
25.5, 26.2, 26.5, 27.0, 28.0, 28.7, 28.9, 28.9, 28.9, 29.4, 30.1, 31.6, 34.0,
36.5, 37.3, 39.0, 38.3, 39.4, 39.3, 41.4, 41.4, 41.3, 42.3, 42.5, 42.4, 42.5,
44.6, 45.2, 45.5, 46.0, 46.6, 34.8, 37.8, 38.8, 39.8, 40.5, 41.0, 45.5, 45.5,
45.8, 48.0, 48.7, 51.2, 55.1, 59.7, 64.0, 64.0, 68.0, 10.8, 11.6, 11.6, 12.0,
12.4, 12.6, 13.1, 13.1, 13.2, 13.4, 13.5, 13.8, 15.2, 16.2), w=c(4.02, 4.3056,
4.6961, 4.4555, 5.134, 4.9274, 5.2785, 4.69, 4.8438, 4.9594, 5.1042, 4.8146,
4.368, 5.0728, 5.1708, 5.58, 5.2854, 5.1975, 5.1338, 5.7276, 5.5695, 5.3704,
5.2801, 6.1306, 5.589, 6.0532, 6.09, 5.8515, 6.1984, 6.603, 6.3063, 6.292,
6.7497, 6.7473, 6.3705, 2.268, 2.8217, 2.9044, 3.1746, 3.5742, 3.3516,
3.3957, 3.2943, 3.7544, 3.5478, 3.8203, 3.325, 3.8, 3.8352, 3.6312, 4.1272,
3.906, 4.4968, 4.7736, 5.355, 4.2476, 4.2485, 4.6816, 6.562, 6.5736, 6.525,
2.3265, 2.3142, 2.673, 2.9181, 3.2928, 3.2944, 3.4104, 3.1571, 3.6636,
4.144, 4.234, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, 3.3756, 4.158, 4.3844, 4.0198, 4.5765, 3.977, 4.3225, 4.459,
5.1296, 4.896, 4.87, 5.376, 6.1712, 6.9849, 6.144, 6.144, 7.48, 1.0476,
1.16, 1.1484, 1.38, 1.2772, 1.2852, 1.2838, 1.1659, 1.1484, 1.3936, 1.269,
1.2558, 2.0672, 1.8792), h=c(11.52, 12.48, 12.3778, 12.73, 12.444, 13.6024,
14.1795, 12.67, 14.0049, 14.2266, 14.2628, 14.3714, 13.7592, 13.9129,
14.9544, 15.438, 14.8604, 14.938, 15.633, 14.4738, 15.1285, 15.9936,
15.5227, 15.4686, 16.2405, 16.36, 16.3618, 16.517, 16.8896, 18.957,
18.0369, 18.084, 18.7542, 18.6354, 17.6235, 4.1472, 5.2983, 5.5756,
5.6166, 6.216, 6.4752, 6.1677, 6.1146, 5.8045, 6.6339, 7.0334, 6.55, 6.4,
7.5344, 6.9153, 7.3968, 7.0866, 8.8768, 8.568, 9.485, 8.3804, 8.1454, 8.778,
10.744, 11.7612, 12.354, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, 2.112, 3.528, 3.824, 4.5924, 4.588, 5.2224, 5.1992, 5.6358,
5.1376, 5.082, 5.6925, 5.9175, 5.6925, 6.384, 6.11, 5.64, 6.11, 5.875,
5.5225, 5.856, 6.792, 5.9532, 5.2185, 6.275, 7.293, 6.375, 6.7334, 6.4395,
6.561, 7.168, 8.323, 7.1672, 7.0516, 7.2828, 7.8204, 7.5852, 7.6156, 10.03,
10.2565, 11.4884, 10.881, 10.6091, 10.835, 10.5717, 11.1366, 11.1366,
12.4313, 11.9286, 11.73, 12.3808, 11.135, 12.8002, 11.9328, 12.5125,
```

12.604, 12.4888, 5.568, 5.7078, 5.9364, 6.2884, 7.29, 6.396, 7.28, 6.825, 7.786, 6.96, 7.792, 7.68, 8.9262, 10.6863, 9.6, 9.6, 10.812, 1.7388, 1.972, 1.7284, 2.196, 2.0832, 1.9782, 2.2139, 2.2139, 2.2044, 2.0904, 2.43, 2.277, 2.8728, 2.9322))

Inits:

list(alpha=120,beta1=8,log.sigma=0)

Linearity by Species

The pair plot of Fish Market data shown in the final comparison but color-coded by Species. We see that each species of fish has its own relatively simple distribution for each predictor, except for Perch, which has a bimodal distribution.

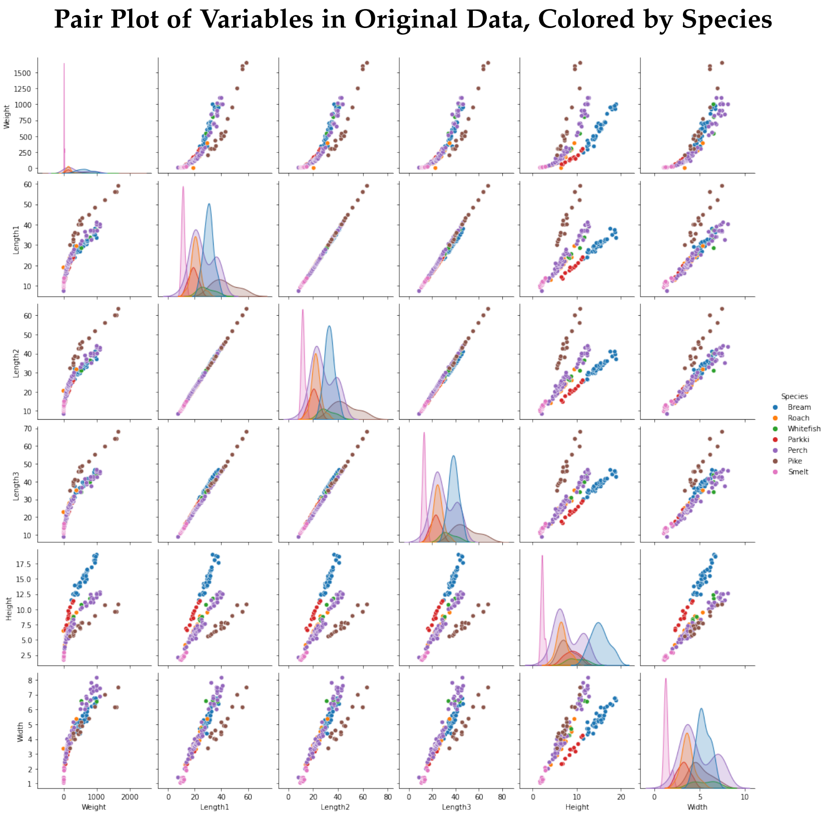


Figure 17: A more detailed look at the distributions and relationships of the variables color-coded by Species.