Section I Project and Review of Key Concepts

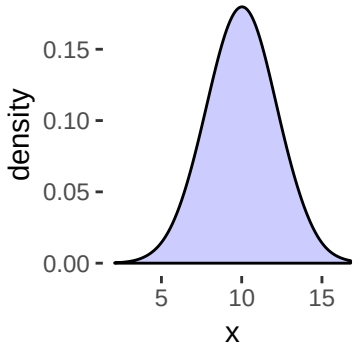
Reviewing Key Concepts from Frequentist Statistics and Machine Learning

As we transition out of frequentist statistics, let's review a few key concepts to keep us grounded while we move on. Particularly, estimation of parameters and confidence.

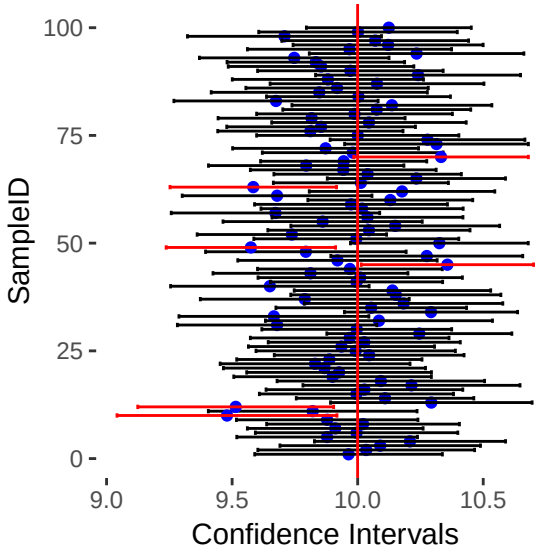
Recall from DA1, that we estimated the parameters of a population of data from samples, and we computed confidence intervals to give us a sense for the reliability of our estimates. For example, when we estimated the mean of a population from samples, we computed the standard error as:

$$\frac{\sigma}{\sqrt{n}}$$

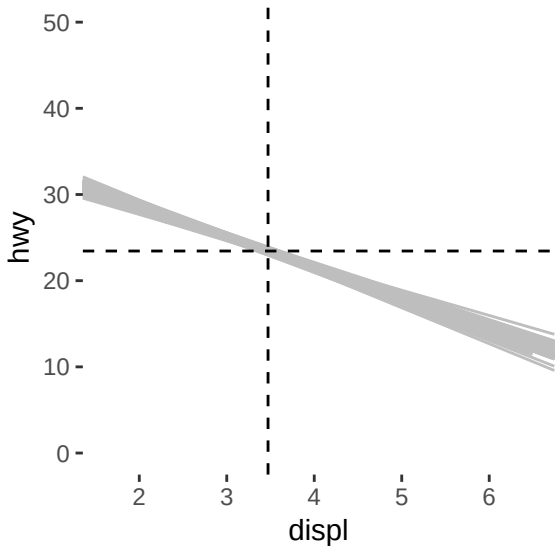
. And to find a 95% **level** we used *qnorm* to calculate a z-score, and multiplied that by the standard error to give us the *confidence interval*. For example, given a population of 10,000 with a mean of 10 and a standard deviation of 2:



We can draw 100 random samples of 100 ea, and compute the confidence *interval* for a *level* of 95%. And not surprisingly, we'll get 95 samples that include the real *(and unknown)* mean.



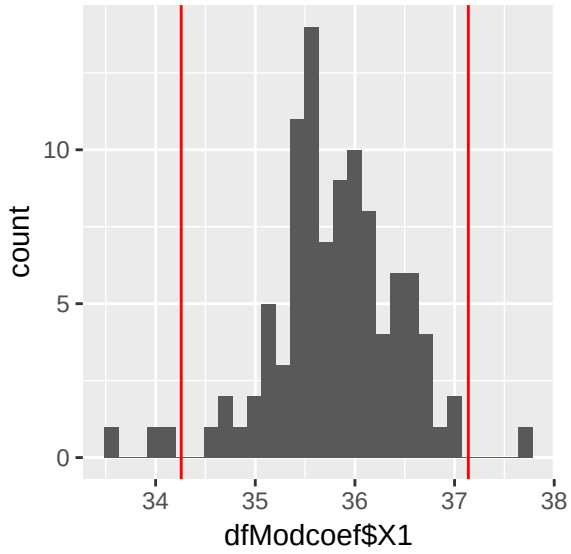
Of course, a mean is just one parameter we might be interested in. We can apply the same confidence intervals *(with some modification)* to other parameters, like a regression intercept. For example, we create 100 linear models of a mpg data:



And we plot the intercept of each model. We said that, in simple models, the confidence interval can be computed as

$$\pm 2 * \sqrt{\frac{SSE}{df}}$$

And not surprisingly, ~ 95% of the estimated sample intercepts fall within that interval



Genearlized Linear Models

As we moved into machine learning, we saw how these principles still applied. With Generalized Linear Models, we still reviewed confidence intervals, but now there are some complicating considerations:

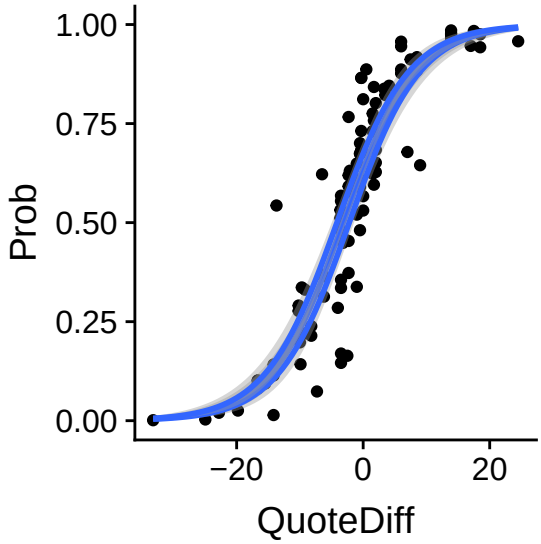
1. The logit link function adds a little complexity: the dependent variable is transformed by the logit function, and so is SE.
2. GLM uses likelihood to determine parameters, and many glm link functions transform the dependent variable to non-linear space

3. Recall from DA1 that SE is determined differently with multivariate models and there are different approaches to determining SE.

Fortunately, glm provides SE values, so you can use something like:

- test$lcl <- test$Prob - testPred$se.fit
- test$ucl <- test$Prob + testPred$se.fit

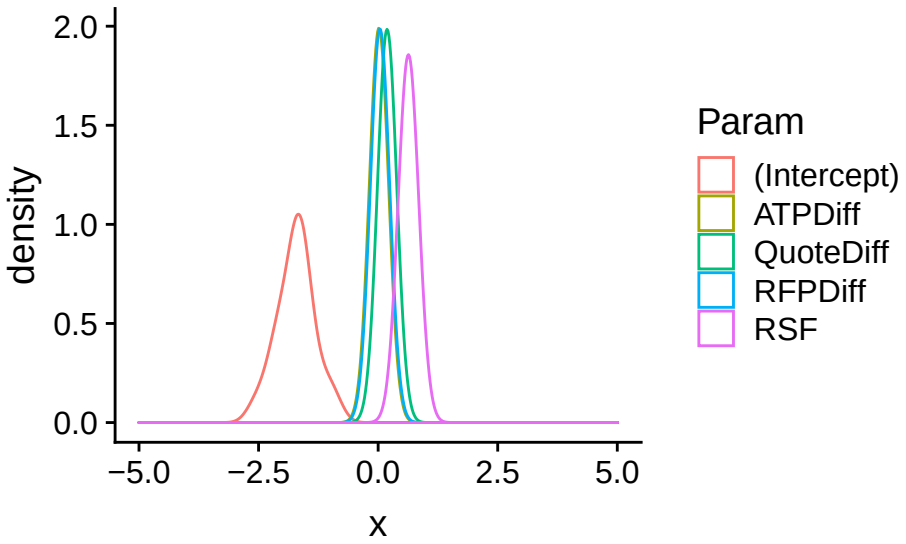
To build you confidence intervals, as shown below:



Confidence intervals for the **independent** variables are on scale, and GLM provides a function to compute these:

- confint(glm.fit) # this uses profile likelihood to compute CIs
- confint.default(glm.fit) # this uses likelihood to compute Wald CIs - we'll use Wald here *(traditional symmetric method)*

Using these CIs, I simulated the parameter distributions shown below *(see R file for method)*:



These intervals give us a metric for quantifying our *confidence* in the model :).

Recap and Looking Forward

These frequentist metrics are widely applicable and will work in many scenarios. There are differences in the approach of frequentist and Bayesian *(which is the basis of the second half of the course)*. Consider the follwoing points:

- For **frequentists**, the parameters are fixed but unknown and the data are random. Again, review how we estimated the parameters *(the mean of the distribution, the intercpept of the linear model, all of the parameters of the generalized linear model - all of these are fixed, we're just trying to figure out what they are by taking samples of the population)*. Frequentist processes fit well when the population size is unknown *(e.g., a survey of likely voters, a trail for new drug)*, and we have no experience with the effects *(Bayesian processes include experience, or beliefs, in models - much more on this later)*. If a frequentist model is not working, the solution is to redesign the **sampling process**.
- For **Bayesians**, the parameters are random and the data are fixed. Think back to our study of *Maximum Likelihood* estimation, when we held the data fixed and varied the parameters, measuring the likelihood of each parameter value *(choosing the most likely value)*. We could do that because we held that data fixed *(or else it would have been a total cluster :))* Bayesian modeling produces analyses like the glm parameter analysis above *(this is not often done in a frequentist process - we just test significance and move on)*. We will get into the nuances in section II. If a Bayesian model is not working, the solution is to redesign the **modeling process**.

My personal opinion favors Bayesian modeling in most business transaction environments *(and in my sarcastic view of the world, Frequentists are people with hammers looking for nails, who often reject common sense based on the "scientific method" they learned in middle school)*. There are numerous reasons, including:

1. In transaction environments, the data are usually fixed *(within defined time periods)* and the challenge is to find a model that fits the data *(that is Statistically valid and parameteric, to support inference)*. Sampling is usually not problematic nor expensive.
2. Inference *(i.e., projection)* in Bayesian models is inherent as predictions generate simulation with all the parmeters and confidence *(or credible)* metrics.
3. In Business environments, we usually have some experience with the data, and that experience can be quantified, integrated and modeled in Bayesian processes. Often times, experience *(or beliefs)* can be more informative than data *(especially in "young" or "incomplete" data)*
4. Transaction environments are most often hierarchical *(or multi-level)*, and the effects of levels have varying influence on other levels. Bayesian modeling adapts well to multi-level modeling *(frequentist models for multi-level, mixed effects scenarios exist, but they are not flexible enough for transacation envionments)*
5. Transaction environments are high volume and dynamic, and models become irrelevant quickly. Bayesian models can be architected so that parameters can be constantly passed between analysis / training modes and models in operations *(BIG deal in fintech, and becoming a big deal everywhere else)*

Hopefully, these points are motivating you to begin Section II. But before we do, let's finish the GLM project:

Project Outline

You have been assigned a project to build and compare two-class models in AML and R *(using GLM)*. Your report should be in R Markdown *(RMD)*, with a minimum of the following sections:

1. Description of your modeling process in AML with analysis of results. Model selection is your choice. Model development / training should include resampling and tuning. For 2 class models, an overall accuracy of 80% is a good target, but cases where a minority class presents challenges will be considered. The RMD file should include a narrative with references to the model in AML *(be sure and save your*

model to the class workspace). Also, any relevant analysis plots to support your narrative shoudl be in the RMD.

2. GLM model with GLM predictions and **validation using a LR equation** *(there should be no difference in the predictions - see exercise V2).* Your Analysis should summarize the modeling process with findings including Comparison/Contrasting to the AML model.
3. Analysis of GLM parameters with Confidence Intervals. Just use the same methodology as presented in this document and the updated file *(DA_LogReg_Exercise_V2).*

Presentation. PPTX decks are optional - you can just walk through the paper if you like. Presentations will take place in class 2/28.