Out of sample generalization

Mark Andrews
Psychology Department, Nottingham Trent University

mark.andrews@ntu.ac.uk

Predicting the future

- If we have a model M_1 with parameters θ_1 whose maximum likelihood estimator is $\hat{\theta}_1$, and if the observed data is y_{obs}, then the model likelihood of M_1 is

$$P(y_{obs}|M_1, \hat{\theta}_1).$$

- We can assume that y_{obs} is a sample from the *true generating model* M_{true}.
- How well does M_1 with $\hat{\theta}_1$ predict y_{new} from M_{true}?
- Out of sample predictive performance:

$$\int P(y_{new}|M_1, \hat{\theta}_1)P(y_{new}|M_{true})\,dy_{new}$$

Expected log predictive density

- If we have n independent samples of data from M_{true}, we can calculate

$$\text{elpd} = \sum_{i=1}^{n} \log P(y_i^{\text{new}} | M_1, \hat{\theta}_1).$$

Cross validation

- Rather than waiting for new data to be collected, a simple solution is to remove some data from the data that is used for model fitting, fit the model with the remaining data, and then test how well the fitted model predicts the reserved data.
- This is known as *cross-validation*.
- One common approach to cross-validation is known as K-fold cross-validation.
- The original data set is divided randomly into K subsets.
- One of these subsets is randomly selected to be reserved for testing.
- The remaining $K - 1$ are used for fitting and the generalization to the reserved data set is evaluated.
- This process is repeated for all K subsets, and overall cross validation performance is the average of the K repetitions.
- One extreme version of K-fold cross-validation is where $K = n$ where n is the size of the data-set.

Cross validation

- For leave one out cross-validation, the procedure is as follows.
- Assuming our data is $\mathcal{D} = y_1, y_2 \dots y_n$, we divide the data into n sets:

$$(y_1, y_{\neg 1}), (y_2, y_{\neg 2}), \dots (y_i, y_{\neg i}) \dots (y_n, y_{\neg n}),$$

  where y_i is data point i and $y_{\neg i}$ is all the remaining data except for data point i.
- Then, for each i, we fit the model using $y_{\neg i}$ and test how well the fitted model can predict y_i.
- For each i, we calculate $\log P(y_i|\hat{\theta}^{\neg i})$, which is the logarithm of the predicted probability of y_i based on the model with parameters $\hat{\theta}^{\neg i}$.
- This then leads to

$$\text{elpd} = \sum_{i=1}^{n} \log P(y_i|\hat{\theta}^{\neg i}).$$

Cross validation

```
housing_df <- read_csv(here("data/housing.csv"))
m1 <- lm(log(price) ~ 1, data = housing_df)

i <- 42
m1_not_i <- lm(log(price) ~ 1,
               data = slice(housing_df, -i)
)

mu <- coef(m1_not_i)
stdev <- sigma(m1_not_i)

y_i <- slice(housing_df, i) %>% pull(price)
dnorm(log(y_i), mean = mu, sd = stdev, log = T)
#> [1] 0.03483869
```

AIC

- Cross-validation is an excellent method of model evaluation because it addresses the central issue of out-of-sample generalization, rather than fit to the data, and can be applied to any models, regardless of whether these models are based on classical or Bayesian methods of inference.
- On the other hand, traditionally, cross validation has been seen as too computationally demanding to be used in all data analysis situations.
- One still widely used model evaluation model, the Akaike Information Criterion (AIC), originally proposed by Akaike (1973), can be justified as a very easily computed approximation to leave one out cross validation (see Stone 1977; Fang 2011). AIC is defined as follows.

$$\begin{aligned} \text{AIC} &= 2k - 2\log P(\mathcal{D}|\hat{\theta}), \\ &= 2k + \text{Deviance}, \end{aligned}$$

where k is the number of parameters in the model.

AIC

```
m1 <- lm(log(price) ~ 1, data = housing_df)
m0 <- lm(price ~ 1, data = housing_df)
k <- 2
aic_1 <- as.numeric(2 * k - 2 * logLik(m1))
aic_0 <- as.numeric(2 * k - 2 * logLik(m0))

c(aic_1, aic_0)
#> [1]   472.599 12682.711
```

AIC

- A model's AIC value is of little value in itself, and so we only interpret differences in AIC between models.
- Conventional standards (see, for example, Burnham and Anderson 2003, chap. 2) hold that AIC differences of greater than between 4 or 7 indicate clear superiority of the predictive power of the model with the lower AIC, while differences of 10 or more indicate that the model with the higher value has essentially no predictive power relative to the model with the lower value.

AICc

- A correction for small samples size (i.e. where $n/k < 40$) is

$$\text{AICc} = \text{AIC} + \frac{2k(k+1)}{n-k-1}$$

```
aic_c <- function(model){
  K <- length(coef(model)) # for glm
  N <- nrow(model$model)
  AIC(model) + (2*K*(K+1))/(N-K-1)
}
```

AIC

```
library(splines)
gssvocab_df <- read_csv(here('data/GSSvocab.csv'))

df_seq <- seq(3, 30) %>% set_names(.,.)

M_gssvocab <- map(df_seq,
                  ~lm(vocab ~ ns(age, df = .),
                      data = gssvocab_df)
)

aic_results <- map_dbl(M_gssvocab, aic_c) %>%
  enframe(name = 'df', value = 'aic') %>%
  mutate(df = as.numeric(df))
```

Akaike weights

- We can use *Akaike weights* for model averaging.
- First, we define Akaike *deltas*. For example, for model k of K, we have

$$\Delta_k = \text{AIC}_k - \text{AIC}_{\text{min}},$$

  where AIC_{min} is the model with the minimum AIC value.
- Then, the Akaike weights are defined as

$$w_k = \frac{\exp\left(-\frac{1}{2}\Delta_k\right)}{\sum_{k=1}^{K} \exp\left(-\frac{1}{2}\Delta_k\right)}.$$

- Each w_k can interpreted as the probability that model k has the best out of sample generalization.

Ridge regression

- Ridge regression is a method to reduce variance in estimators of regression coefficients.
- It penalizes large coefficients and shrinks the towards zero.
- In linear regression, it estimates the coefficients by minimizing the penalized sum of squared residuals:

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{k=0}^{K} \beta_k^2,$$

where

$$\hat{y}_i = \beta_0 + \sum_{k=1}^{K} \beta_k x_{ki},$$

and λ is a regularization parameter.

Lasso

- Least absolute shrinkage and selection operator (lasso) is a method similar to ridge regression, but uses a penalty based on the sum of the *absolute* values of coefficients:

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{k=0}^{K} |\beta_k|.$$

Elastic net

- Elastic net is a method similar to ridge regression and lasso.
- It uses as weighted average of the two penalty methods:

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda\left(\alpha\sum_{k=0}^{K}|\beta_k| + (1-\alpha)\sum_{k=0}^{K}\beta_k^2\right).$$

- The values of α ranges from 0 to 1.
- When $\alpha = 0$, this is pure lasso regression.
- When $\alpha = 1$, this is pure ridge regression.

References

Akaike, Hirotogu. 1973. "Information Theory and an Extension of the Maximum Likelihood Principle." Edited by F Petrov B. N AND Caski.

Burnham, Kenneth P, and David R Anderson. 2003. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer Science & Business Media.

Fang, Yixin. 2011. "Asymptotic Equivalence Between Cross-Validations and Akaike Information Criteria in Mixed-Effects Models." *Journal of Data Science* 9 (1): 15–21.

Stone, M. 1977. "An Asymptotic Equivalence of Choice of Model by Cross-Validation and Akaike's Criterion." *Journal of the Royal Statistical Society. Series B (Methodological)* 39 (1): 44–47.