Network models - Class October 10

Scribe: Lynn Kaack

October 12, 2016

1 Agenda

Link prediction / CV
- multi-fold CV
+ "random" vs. "net" vs. "Latin"
+ strengths and limits
- leave-one-out
Goodness of fit
-Simulation-based
- choice of test stats.
- "spectral goodness of fit"
+ Laplacian
+ Why laplacian?

2 Link Prediction

Link Prediction = cross-validation

Goodness of fit & specification checking

Link prediction: given data on part (most) of the network, predict whether there is an edge between given pair of nodes, i.e. does $A_{ij} = 1$?

- Classification problem
- Can give probabilities, $Pr(A_{ij} = 1)$
  - most models in this class will naturally give $Pr(A_{ij} = 1)$, e.g. in stochatisc block model get $Pr(Z_i = r, Z_j = s)$ and use b_{rs} with those weights.

- Do not try to predict the data you used to fit the model
  - This just encourages over-fitting, i.e. memorizing noise in the fitting data
  - Do not try to "explain variance"
- Predict new data, not used in fitting
  - but totally new data is very expensive
- $\star$ Cross-validation: pretending some data is new, then swapping roles

3 How do we divide the data for CV?

For IID data: Leave-one-out vs. k-fold

Leave-one-out is generally: leave one node pair ("dyad") out, fit the model using all pairs *except* (i,j), predict $Pr(A_{ij} = 1)$, evaluate our loss on (i,j), average over all $n(n-1)$ (or $\binom{n}{2}$) pairs

Problem: Leave-one-out isn't consistent for model selection even for IID data

LOOCV over-fits with a prob $\rightarrow p > 0$ even as $n \rightarrow \infty$ (AIC is a rough approximation to LOOCV)

K-fold CV is model-selection consistent (for IID data, with some ceveat).

3.1 How do we do k-fold CV for networks?

"Random CV": assign all dyads at random to the k-folds
Chen & Lei 2014 "net CV" (refer to Fig. 1)

- Assign nodes to k folds randomly and in equal numbers
- The test set for fold f is all pairs between nodes in that fold

Dabbs & Juner 2016: "Latin" CV (refer to Fig. 2)

- Edges are divided into k folds
- Each edge to be in on of the folds
- Start with an arrangement of numbers $l : k$ on $n \times n$ grid, such that each $f \in l : k$ appears an equal number of times in each row and each column
- Permute while maintaining balance for folds across rows and columns

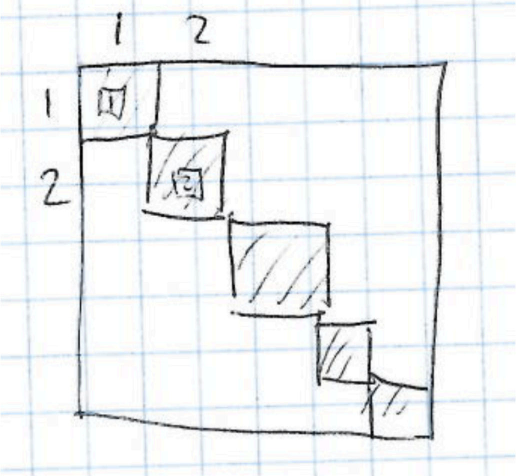


Figure 1: net CV illustration

To decide on which method to do, we cannot rely on any theory (yet). Numerically, random and "Latin" seem to work better at model selection and prediction risk estimation than net CV. Latin is less variable (at small n) than random.

4 Evaluating the predictions

Classification accuracy is ok but ignores the probabilities.

4.1 Proper scoring rules

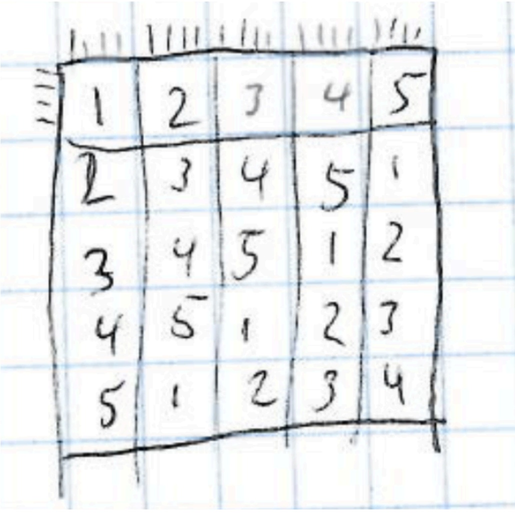
Proper scoring rules = loss functions for comparing $Pr(Y = 1)$ to Y s.t. minimum risk is only happening at the true distribution.

For binary outcomes, one proper scoring function is $(Y - Pr(Y))^2$ (Brier Score).

For general outcomes, "log loss", $-\log Pr(Y)$, is proper. The expectation is $-\sum_y Pr(Y = y) \log(Pr(Y = y)) = entropy$ (if using true distribution). If truth is Q but we predict P, $H[Q] + D(Q||P)$.

4.2 Calibration

Calibration: probability forecast is *calibrated* if events forecasted to have probability p happen p% of the time.



1	2	3	4	5
2	3	4	5	1
3	4	5	1	2
4	5	1	2	3
5	1	2	3	4

Figure 2: Latin CV illustration

Calibration is weaker than getting the distribution right. For example we have HTHTHTHTHTHT.... but we predict: $Pr(H) = 0.5$. Helpful plot is in Fig. 3.

5 Goodness of fit

Does the data look like a typical outcome of the generative model?

Model predicts some regular trends and fluctuations around trends, are the data's deviations from trend typical fluctuations (per model)?

E.g. in regression: no trends in residuals vs. predictors, constant variance, Gaussian marginal, no correlation, etc.

For networks, we usually do all this by simulation (Fig. 4 and 5)

- Fit the model to data
- Simulate many networks from the model
- Calculate test statistic(s) on each simulated network
- Ideally, observed test statistics are typically of similar values

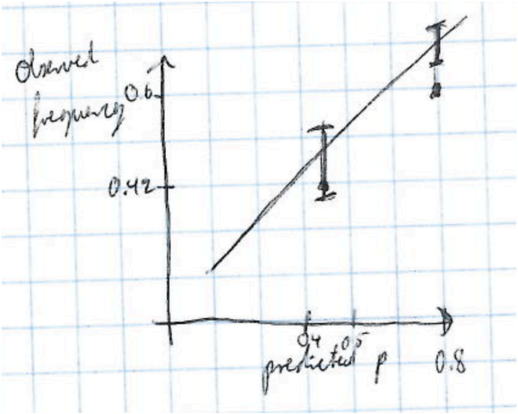


Figure 3: A helpful plot for calibration.

5.1 Choice of test statistics

In principle you can use any function of the data.

Typically, low power for statistics directly used to fit the model

- e.g. sufficient statistics for an ERGM (degeneracy is a situation with very high power)
- or statistics which are mathematically dependent on them

Prefer statistics which are not involved in fitting, easy to calculate, give some info about interesting graph properties. EDA or dynamical properties or properties or competing models.

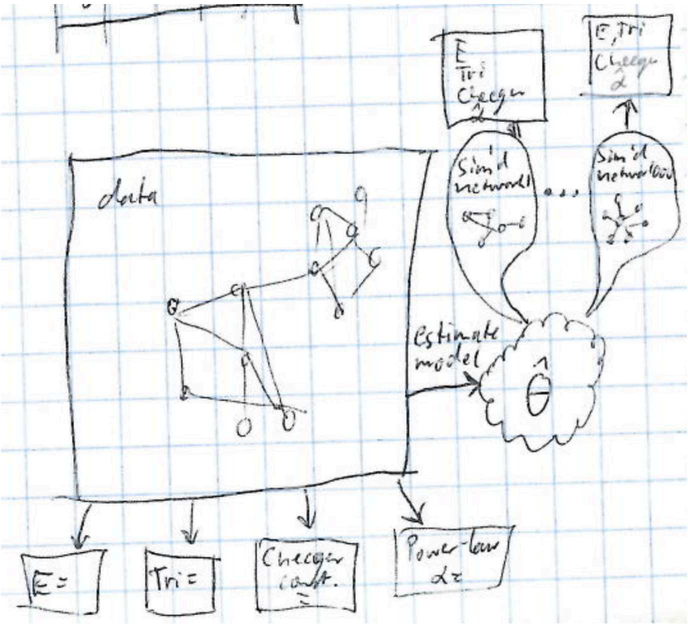


Figure 4: How to test with simulations if the data's deviations are typical.

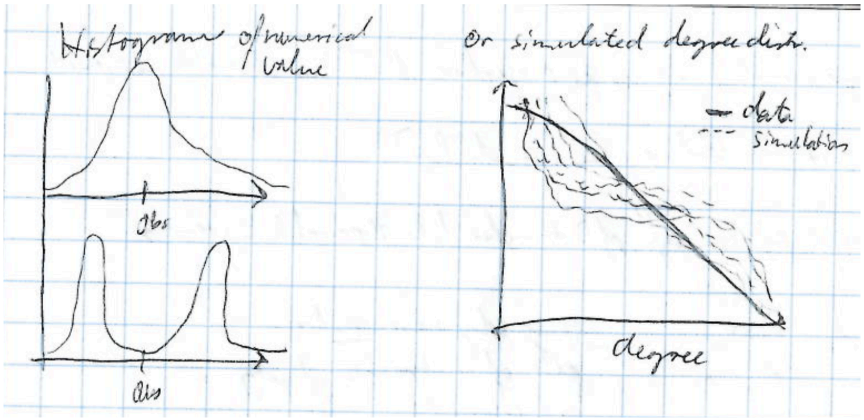


Figure 5: Simulations vs. data