Confidence sets for binary classification problems

Noémi Takács

Supervisor: Ambrus Tamás, SZTAKI, ELTE

1 Introduction

In the previous semester I empirically analysed confidence intervals for mean estimation problems, where I compared the performance of the sign-perturbed sums (SPS) algorithm to asymptotic methods. I also examined the simplest "classification problem" in which case there are no explanatory variables.

In the second part of the project I considered binary classification with one explanatory variable, using the generalization of the SPS method [1, 2]. The aim is to estimate the regression function (f_*) and construct confidence regions around the estimate. In this report first, I present the SPS method for classification, then I make simulations with generated mixed Laplace distributed random variables and finally apply the method on real data associated with bank churn.

In binary classification an i.i.d. sample is given from the unknown distribution of a random vector variable $(X, Y) \in \mathbb{X} \times \mathbb{Y}$, where $\mathbb{X} \subseteq \mathbb{R}^d$ is the input space and $\mathbb{Y} = \{1, -1\}$ is the binary output space. The (measurable) $g : \mathbb{X} \to \mathbb{Y}$ functions are called classifiers. Generally the objective is to find a Bayes optimal classifier, which minimizes the Bayes risk $R(g) \doteq E[L(Y, g(X))]$, where L is a nonnegative measurable loss function. If $L(Y, g(X)) = \mathbb{I}(Y \neq g(X))$, where $\mathbb{I}$ is the indicator function, then the Bayes optimal classifier will be the sign of the regression function, $f_*(x) \doteq E[Y|X = x]$. The role of the regression function in binary classification is central, because it determines the conditional probabilities of the classes given the input as

$$f_*(x) = E[Y|X = x] = 2 \cdot P(Y = 1|X = x) - 1.$$

If the corresponding densities exist:

$$P(Y = 1|X = x) = \frac{P(Y = 1) \cdot f_{X|Y=1}(x)}{P(Y = 1) \cdot f_{X|Y=1}(x) + P(Y = -1) \cdot f_{X|Y=-1}(x)}.$$

Although in practice we do not know $f_{X|Y=1}$ and $f_{X|Y=-1}$, we can choose distribution families and approximate them. Thus I will construct confidence sets for the parameters and the regression function in a given model class.

2 Confidence sets

I empirically analysed the SPS method for classification problems, which is a resampling procedure. It can construct exact confidence sets with a user-chosen confidence level for the regression function under the following mild assumptions:

(a1) $\mathbb{X} \subseteq \mathbb{R}^d$ and the $\{(X_j, Y_j)\}_{j=1}^n$ sample is i.i.d.;

(a2) for the regression function a parameterised family $\mathcal{F}$ is given, which contains f_*, i.e.,

$$f_* \in \mathcal{F} \doteq \{f_\theta : \mathbb{X} \to [-1, 1] \,|\, \theta \in \Theta\};$$

(a3) the parameterisation is injective, such that for all $\theta_1 \neq \theta_2 \in \Theta$:

$$\|f_{\theta_1} - f_{\theta_2}\|_P^2 \doteq \int_{\mathbb{X}} (f_{\theta_1}(x) - f_{\theta_2}(x))^2 \mathrm{d}P_X(x) \neq 0,$$

where P_X is the distribution of the inputs.

Without the second one, it would not be possible to construct the set in practice and the third one is a technical assumption.

2.1 Resampling framework

The main idea is to generate, for a given θ parameter, $m-1$ alternative outputs for the original inputs from the conditional distribution given θ:

$$\mathbb{P}_\theta(Y = 1|X = x) = \frac{1 - f_\theta(x)}{2}.$$

Let $\mathcal{D}_0 = \{(X_j, Y_j)\}_{j=1}^n$denote the original sample, then we construct the i-th alternative sample by

$$\mathcal{D}_i(\theta) \doteq \{(X_j, Y_{i,j}(\theta))\}_{j=1}^n,$$

where $Y_{i,j}(\theta)$ is generated from $\mathbb{P}_\theta(Y = 1|X_j)$. Here we have two remarks:

1. If $\theta = \theta^*$, then $\mathcal{D}_0$ and $\mathcal{D}_i(\theta^*)$ comes from the same distribution.

2. If $\theta \neq \theta^*$, then the distribution of $\mathcal{D}_i(\theta)$ differs from that $\mathcal{D}_0$.

The significance of the difference can be detected with a statistical test, considering the following hypotheses:

$$\begin{aligned} H_0 &: f_* = f_\theta \\ H_1 &: f_* \neq f_\theta \end{aligned} \tag{1}$$

The original and alternative samples are compared using a ranking function.

2.2 Compute ranks

In the following I determine the rank for a concrete distribution family and a given θ as follows:

1. Determine $f_{\hat{\theta}}$, where $\hat{\theta}$ is the maximum-likelihood estimate of the parameter.
2. Calculate
$$Z_0(\theta) = \frac{1}{n}\sum_{j=1}^{n}\left(f_{\hat{\theta}}(X_j) - f_\theta(X_j)\right)^2,$$
where X_j is the j-th input element.
3. Generate $m-1$ alternative outputs, and determine $f_{\hat{\theta}_i}$, where $\hat{\theta}_i$ is the ML-estimation of θ from the i-th new sample for $i = 1, \ldots, m-1$.
4. Calculate
$$Z_i(\theta) = \frac{1}{n}\sum_{j=1}^{n}\left(f_{\hat{\theta}_i}(X_j) - f_\theta(X_j)\right)^2,$$
for $i = 1, \ldots, m-1$.
5. Order $\{Z_i\}_{i=0}^{m-1}$ and compute the rank by:
$$\mathcal{R}(\theta) \doteq 1 + \sum_{i=1}^{m-1}\mathbb{I}\left(Z_i(\theta) \prec_\pi Z_0(\theta)\right),$$
where $\prec_\pi$ is a strict ordering, cf. [1]
6. Return $\mathcal{R}(\theta)$.

Table 1: Pseudocode: determine rank

Definition 1 *Let $\mathcal{R}$ be as in the pseudocode and $p \leq q \in [m]$ user-chosen integers, then the SPS confidence set is defined by*

$$\Theta_\varrho \doteq \{\theta \in \Theta : p \leq \mathcal{R}(\mathcal{D}_0, \{\mathcal{D}_k(\theta)\}_{k\neq 0}) \leq q\},$$

where $\varrho \doteq (m, p, q)$.

Theorem 1 *Assuming (a1), (a2) and (a3), for ranking function $\mathcal{R}$ and $\varrho \doteq (m, p, q)$ parameters for which $1 \leq p \leq q \leq m$,*

$$\mathbb{P}(\theta^* \in \Theta_\varrho^\psi) = \frac{q - p + 1}{m}.$$

This theorem guarantees non-asymptotically the exact inclusion probability of f_* under mild statistical assumptions. It is independent from the distribution of the inputs, hence it is semi-parametric.

The test for problem (1) could be: accept the nullhypothesis if the tested regression function is in the confidence set. Then theorem 1 determines exactly the significance level of the test, which we can set arbitrary.

3 Simulations

I made some simulations on generated data, where $P(Y = 1) = P(Y = -1) = 0.5$ and the conditional distribution of X given Y are Laplacian with location parameter $\mu = Y$. The scale parameter λ equaled to 1 in both of the distributions. In this case the (real) conditional probability is:

$$P(Y = 1|X = x) = \frac{e^{-|x-1|}}{e^{-|x-1|} + e^{-|x+1|}}.$$

In all my analyses I fixed μ_1 and μ_2 as known, and I tested parameters $p = P(Y = 1)$ and λ to produce figures in 2 dimension.

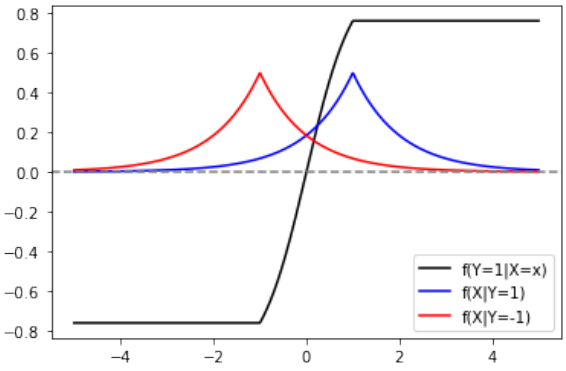

Figure 1: The two Laplace PDF and the real regression function

I implemented the algorithm and first generated a sample with 500 elements. Then I set m to be 20 and tested $51 - 51$ parameters for p and λ uniformly from the intervals $[0.4, 0.575]$ and $[0.78, 1.25]$ in all combinations. The results in the

parameter and model space are shown in Figure 2. Color black denotes the 5 % confidence level and as the orange becomes lighter, the confidence level increases to 100 %. Color aqua denotes the real parameters and regression function used to generate the sample.

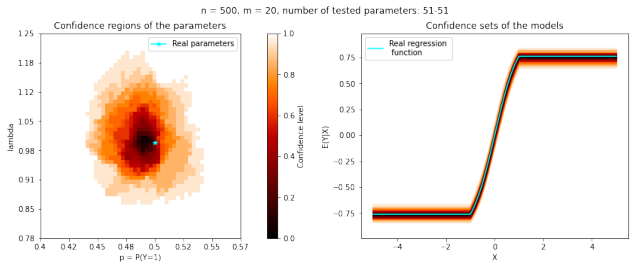

Figure 2: Confidence sets for $n = 500$ in the parameter and the model space

In the second experiment with mixed Laplace sample I made some changes. I set m to be 10 and the sample size from 20 to 300. I ran the code in two different ways as a comparison: first with the known μ_1 and μ_2 and for the second time always with the ML estimation of the location parameters from the actual sample. I repeated the previous steps five times and illustrated the mean of the ranks and the related regression functions (Figure 3). The generated samples are pairwise the same. The first two rows belong to the tests with known μ_1 and μ_2, and one can see the results of the estimated location parameters on the second half of the graphics.

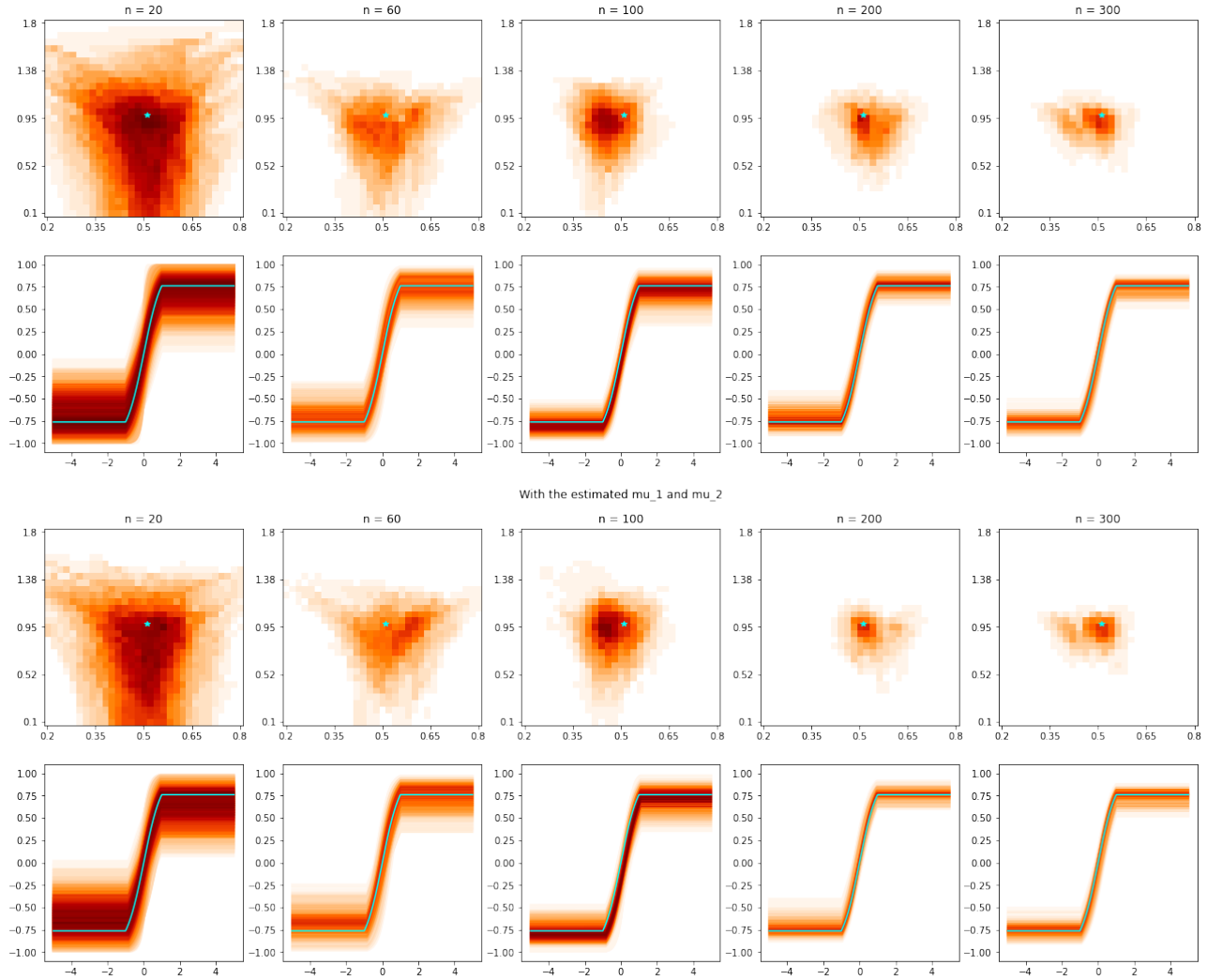

Figure 3: Confidence sets for different sample sizes and multiple running to compare the results for known and estimated location parameters. Color aqua candidates the real parameters and regression function that we used to generate the sample.

We can conclude that a smaller sample size induces more uncertainty (it is not surprising). But we can say that there is just a bit difference if we use the estimated parameters instead of the real ones. This is good because in general we usually do not even know the distribution families, let alone their true parameters.

4 Application to real data

In this section I apply the presented method on real data. The used dataset [3] is publicly available on Kaggle. It contains information about customers of a bank, and the aim is to predict the probability that someone will leave this institute. In the dataset output variables 1 and 0 denote the fact of churn and staying. Many features are available about the clients, from which I selected age as the explanatory variable. Because there were so many observations in the database, I reduced the size to 4000 to keep our methods in a tractable regime. I randomly selected the clients, keeping the original exited - not exited ratio. The histogram of the observations is shown in Figure 4.

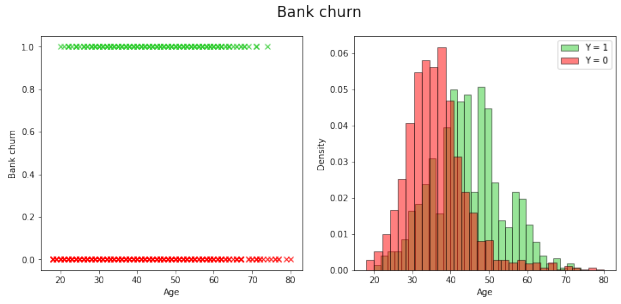

Figure 4: Histogram of the observations

We can see that the distribution of the ages, where people leaved the bank is similar to a normal, but the age distribution of the other group of people is clearly skewed, and has a higher kurtosis than gaussian distribution. In my first experiment I chose two model class for constructing confidence sets: firt I approximated both of the distributions with normal, then with lognormal distributions, which is skewed. Similarly to the Laplacian case, I fixed μ_1, μ_2 and σ_1^2 with ML-estimates, which are the expected values and variance of exited at normal and the same parameters of the logarithm of the observations at lognormal distribution. Thus I tested the $p = P(Y = 1)$ and σ_2^2 parameters. My additional hyperparameters: $n = 4000$ (all), $m = 10$, number of tested parameters $= 21 - 21$. The results in the parameter and model space can be seen in Figure 5. Color aqua denotes the ML estimate of the tested parameters and regression functions.

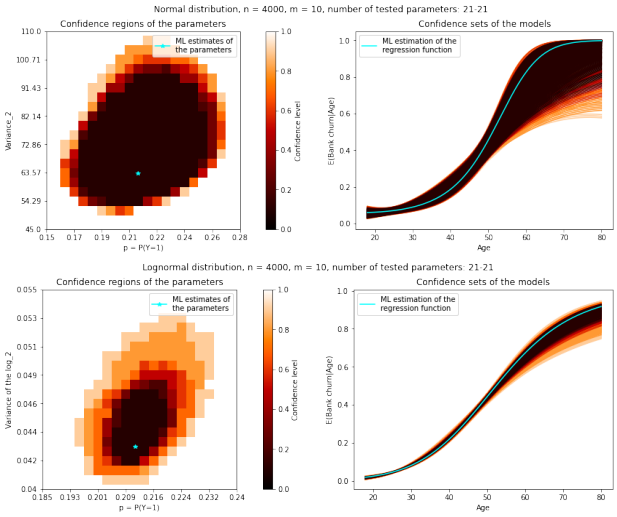

Figure 5: Predicting bank churn using gaussian (above) and lognormal (below) distribution

One can conclude that although predicting exited = 1 seems to be a harder exercise, the model class using the lognormal distribution is much more confident than the other. In Figure 6 I chose some age values and made plots of the confidence intervals of the probability that a client at these ages will leave the bank. As the illustrations show, the length of the intervals given by the lognormal class were always shorter, but the two case are not in conflict, the longer interval usually contains the shorter one.

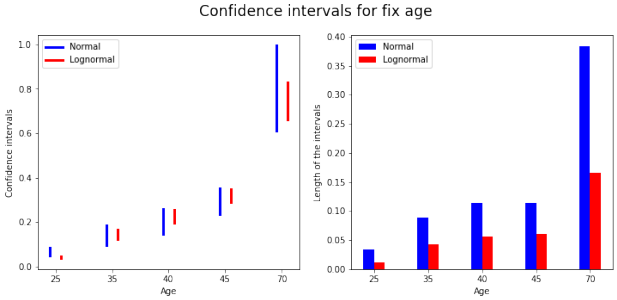

Figure 6: Confidence intervals and their length for the two model classes

My last study used the model class with log-normal distribution. In this case I took smaller samples of 200 elements, repeated 5 times and illustrated the mean of the ranks. The 5 samples were element-wise different. The number of tested parameters were $30 - 30$ and m was set to 10. I made two versions to compare the confidence sets in the model space:

1. test p and σ_2^2

2. test p and μ_2

The plots are shown in Figure 7.

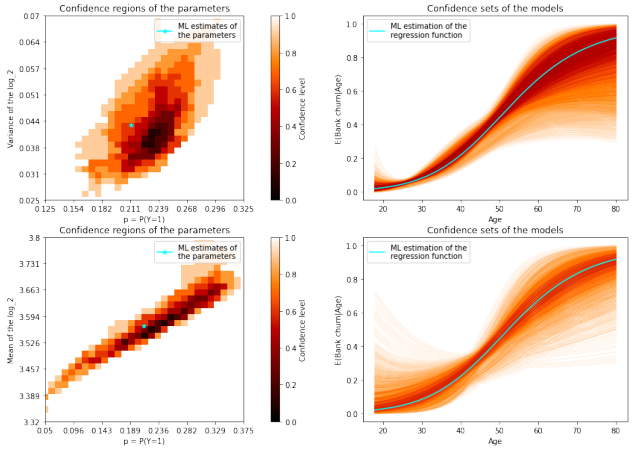

Figure 7: Test μ_2 vs. σ_2^2: confidence sets with $n = 200$ from multiple sampling

We can say that there is a difference between the results of the two tests. Both cases indicated a bigger challenge in predicting the outcome at higher ages, but changing the location parameter for people staying at this bank also led to longer confidence intervals for the probability of churn at lower ages. As one can see on figure 4, there are observations from both outputs among the youngest, not only close to the maximum age in this database. Hence, there may be higher uncertainty at the two edges of the interval of possible input values.

5 Conclusions

In this report I presented a non-asymptotic, distribution-free, and exact confidence set constructing algorithm for binary classification problems. I also demonstrated how the method works via synthetic and real data.

This method can be very useful, especially when we have a small sample, because it requires mild statistical assumptions. There is also an open question. Testing more parameters makes it difficult to represent the confidence sets in a higher dimensional parameter space. Of course we can give a yes/no answer if the set contains the tested parameters in a user-chosen level, but it is hard to visualize it in higher dimension, and also the computational time increases.

In the next semester the aim is to study this method in multivariate cases and find an effective way to represent the results. One of the main objectives is to extract confidence intervals from the sets in the case of more parameters and higher dimension. Taking more variables into the model can give better predictions in classification problems, e.g. in this bank churn dataset.

References

[1] Tamás, A., Csáji, B. Cs. (2020). Sztochasztikus garanciák bináris klasszifikációhoz. *Alkalmazott matematikai lapok*, 37, 365–379.

[2] Tamás, A., Csáji, B. Cs. (2022). Exact Distribution-Free Hypothesis Tests for the Regression Function of Binary Classification via Conditional Kernel Mean Embeddings. *IEEE CONTROL SYSTEMS LETTERS*, VOL. 6, 860 - 865

[3] willian oliveira gibin, and SIMARPREET SINGH. accessed: 04.22.2024. (2024). Bank Churn Prediction [Data set]. *Kaggle* https://doi.org/10.34740/KAGGLE/DSV/7466166