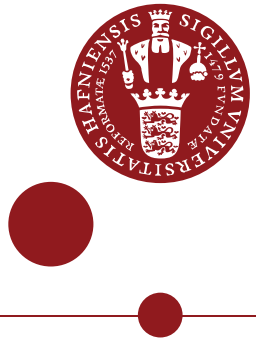
UNIVERSITY OF COPENHAGEN
DEPARTMENT OF ECONOMICS



Seminar paper

Magnus Lindgaard Nielsen, wkg579

Structural breaks in predictive modelling

Detecting structural breaks in prediction or classification problems with repeated cross sections

Seminar: Applications of Machine Learning in Economics, spring 2021

Supervisors: Joachim Kahr Rasmussen and Kristian Urup Olesen Larsen

Length: 35939 characters incl. spaces

Submitted on: 01/06/2021

Abstract

A novel method to detect structural breaks in classification or prediction problems with repeated cross sections is proposed. The theoretical foundation is presented and the power and robustness of the method in a specific use case is showcased with a simulation study. The method utilizes that the generalization error of a model is an IID sample from a distribution which can only change between periods if the data generating process changes. Utilizing holdout data in each period, any changes in performance can then be attributed to a structural break in the data. This is tested utilizing tests for equality of means between the samples.

Contents

1 Introduction

When creating a model to capture the relationship in a data set, it is important to know if the relationship in the data set has changed throughout time. In econometrics, this is known as a structural break. The perhaps most known method of detecting structural breaks is the Chow test developed by Chow in 1960 [1], which test for equality of coefficients in two linear regressions. There also exists a literature within time series forecasting, both trying to pinpoint when structural breaks occur [2][3], but also how to handle structural breaks when doing out-of-sample forecasting [4] [5].

However, neither of the two situations encapsulate prediction or classification problems in repeated cross sections. This is because 1) coefficients are not of specific interest and 2) we are not modelling time series data. To the authors knowledge, no literature exists which has focus on structural breaks in prediction or classification problems. Inspired by the out-of-sample forecast literature in times series econometrics and the widespread use of model performance metrics in the machine learning community, we propose a novel method to detect structural breaks in a data generating process (henceforth DGP) by detecting breaks in the accuracy of a model. In essence, by creating a set of IID generalization errors utilizing holdout data for each cross section, we can compare the models performance for each cross section and hereby detect whether the DGP has changed between periods.

Section 2 describes the theory behind the method for generating the IID sample of generalization errors, followed by a discussion and recommendation of how to detect breaks. Section 3 includes a simulation study, which shows that the method works in practice with a prediction problem solved with the LASSO, and several different setups are utilized to examine the power of the test. Section 4 consists of a discussion of drawbacks and possible further work, which is followed by a conclusion in section 5.

2 Theory

In section 2.1, we first outline how to make inference for the generalization error with a single cross section. In section 2.2, we show how this can be utilized to detect breaks in the DGP when multiple cross sections are available. Section 2.3 concludes with a recommendation on how to handle the multiple testing problem.

2.1 Inference for the generalization error

This subsection will outline inference for the generalization error as it is usually utilized when one is interested in estimating the accuracy of a single decision rule on a given domain. This is done such that the known and common method, which is explained in Nadeau and Bengio (2003) [6], is clearly outlined before we make any modifications to adapt it to our needs. Most of the notation is an adjusted version of Nadeau and Bengio (2003) [6].

In essence, we show that, utilizing a holdout data set, we have unbiased estimators of the mean and variance of the models performance on the given DGP. This is widely used to compare different algorithms predictive power and is the cornerstone of cross validation.

It is important to note that this paper aims only to make inference for the generalization error of a specific decision rule, and thus neither the generalization error for an infinite amount of folds utilizing some form of cross validation or of a learning algorithm in and of it self. According to Dietterich's taxonomy, this places us squarely in the statistical question 1, i.e. how to predict a models accuracy on a given domain with a sample large enough to create a holdout data set [7]. This is an important distinction to make, as we are able to estimate the mean and variance of this generalization error without bias, whereas comparing the generalization error of a model when utilizing cross validation, a model across domains or a learning algorithm in and of itself entails different problems which are not pertinent to this method (see [6], [8] and [9] for further work on the generalization error in a broader scope).

The general setup is that we observe data of the form Z_1^n with size n:

$$Z^n = \{Z_1, ..., Z_n\} \tag{1}$$

With each Z_i defined by:

$$Z_i = (X_i, Y_i) \in \mathcal{Z} \subseteq \mathbb{R}^{p+q} \tag{2}$$

Where p and q are the dimensions of X_i and Y_i.

Z_i's are independently and identically distributed with unknown distribution P:

$$Z_i \sim P(Z) \tag{3}$$

To summarize, we have a set of target covariates Y_i and input covariates, X_i, which follow some unspecified distribution, $P(Z)$.

Our object of interest is the *generalization error*, $\mathcal{L}(D, Z_{n+1})$, which is a measure of how well our model performs on data from the unknown distribution $P(Z)$. Let $\mathcal{L}(D, Z_{n+1})$ be a function

from $\mathcal{Z}^n \times \mathcal{Z}$ to $\mathbb{R}$, where D is a subset of Z^n of size $n_D \leq n$ and $Z_{n+1} = (X_{n+1}, Y_{n+1})$ is a draw from $P(Z)$ which is not in D. Expanding on this function:

$$\mathcal{L}(D; Z_{n+1}) = \mathcal{L}(D; (X_{n+1}, Y_{n+1})) = Q(F(D)(X_{n+1}), Y_{n+1}) \tag{4}$$

We see that the generalization error consists of two parts:

- A decision rule $f = F(D)$ based on D, and as such takes input data of dimensions $\mathbb{R}^p$ and outputs a prediction of dimensions $\mathbb{R}^q$ $[F(D) : \mathbb{R}^p \to \mathbb{R}^q]$. In practice, this decision rule will be created by a learning algorithm, however this is not a requisite, and as such could also be a heuristic developed by a human or, in fact, any other arbitrary decision rule.
- An accuracy measure $Q(\hat{Y}_{n+1}, Y_{n+1})$, where $\hat{Y}_{n+1} = f(X_{n+1})$ is the prediction of the decision rule given input X_{n+1}. For regression problems with $q = 1$ this could be the mean squared error and for classification problems it could be the indicator function indicating whether the classification was correct. This flexibility allows the researcher to utilize any accuracy measure that the researcher is interested in and detects breaks in this specific accuracy measure.

In essence, we utilize our development data D to create a decision rule $F(D)$ (e.g. select hyperparameters and fit our model to the data) and we then measure the accuracy on data not in D utilizing some accuracy measure (e.g. mean squared error for prediction or log-loss score for classification).

We are now interested in estimating $\mu_D \equiv \mathrm{E}[\mathcal{L}(D, i)],\ i \notin D$, i.e. the average generalization error of the *specific decision rule* trained on the subset D on unknown data from the same unknown distribution $P(Z)$. What follows equals statistic number one from Nadeau and Bengio (2003) [6].

To achieve this, we utilize a method completely analogous to splitting the data into a development and holdout data set and estimating the performance of the model (on data from the same domain) utilizing the holdout data set.

To obtain a holdout data set, we limit D to be of size $n_D < n$ with $n_H = n - n_D$, and denote the holdout set $H = Z^n \setminus D$. This allows us to compute n_H IID generalization errors for the specific decision rule trained on D. As we have an IID sample, we can ubiasedly estimate the population sample and variance, which allows us to make inference for the generalization error utilizing a central limit theorem.

To estimate the population mean, μ_D, and the population variance, $\mathrm{V}[\mathcal{L}(D,i)] = \sigma_D^2, i \notin D$, we define the estimator $\hat{\mu}_D$, calculated as the sample mean:

$$\hat{\mu}_D \equiv \frac{1}{n_H} \sum_{i \in H} \mathcal{L}(D;i) \tag{5}$$

And the variance estimator $\hat{\sigma}_D^2$, calculated as the sample variance with Bessel's correction:

$$\hat{\sigma}_D^2 \equiv \frac{1}{n_H - 1} \sum_{i \in H} \left(\mathcal{L}(D;i) - \hat{\mu}_D\right)^2 = \frac{1}{n_H - 1} \sum_{i \in H} \left(\mathcal{L}(D;i) - \frac{1}{n_H} \sum_{i \in H} \mathcal{L}(D;i)\right)^2 \tag{6}$$

The specific central limit theorem utilized is the Lindeberg-Lévy central limit theorem as derived in Rao 1973 [10], as we have already assumed that Z_i is IID. Under the further assumptions (or 'regularity conditions') that $\mathrm{E}[\mathcal{L}(D, Z_i)] = \mu_D$ and $\mathrm{V}[\mathcal{L}(D, Z_i)] = \sigma_D^2$ exist, the Lindeberg-Lévy central limit theorem specifies that:

$$\frac{\hat{\mu}_D - \mu_D}{\sqrt{\frac{\hat{\sigma}_D^2}{n_H}}} \xrightarrow{d} N(0,1) \tag{7}$$

for $n_H \to \infty$.

As such, we are able to make inference for the generalization error of a specific decision rule on a given unknown distribution. In practice, this allows a researcher to split up a dataset from an unknown distribution into a development (D) and holdout (H) data set and make valid inference for the generalization error of a decision rule. However, it is important to note that the distribution of H does not need to be the same as D to enable valid inference for the generalization error of a model trained on D.

2.2 Detecting structural breaks

We have now derived how to make inference for the generalization error of a model on a given holdout data set with a single cross section, widely used in model evaluation. We now introduce a setup with repeated cross sections. Each cross section will have it's own holdout data set, and it is in these holdout data sets we will examine whether structural breaks (i.e. changes in performance or changes in the distribution of the generalization error) occur.

We first go through the setup of the data and derivations for the mean and variance estimator. This is followed by a simple three period univariate example, showcasing the procedure. We then recommend a test to compare the generalization error in two period, before tackling them multiple testing problem in section 2.3.

The setup consists of T repeated cross sections, with each period t having size n_t, which is allowed to vary in between periods:

$$Z_t^n = \{Z_{t,1}, ..., Z_{t,n_t}\}, \qquad t \in \{1, .., T\} \tag{8}$$

Where each $Z_{t,i}$ defined by:

$$Z_{t,i} = (X_{t,i}, Y_{t,i}) \in \mathcal{Z} \subseteq \mathbb{R}^{p+q} \tag{9}$$

Where p and q are the dimensions of $X_{t,i}$ and $Y_{t,i}$, which are fixed for all periods.

All $Z_{t,i}$'s are independently distributed and are furthermore also assumed to be identically distributed within periods, with each period t having an unknown distribution P_t:

$$Z_{t,i} \sim P_t(Z_t) \tag{10}$$

This is analogous to simply repeating the previous setup, and allowing $P(Z)$ to change between periods.

To create a decision rule, we restrict a subset S_D of size $n_{S_D} < n_1$ in period 1 (the 'baseline' period) to be used only for development, and utilize the remaining holdout subset S_H of size $n_{S_H} = n_1 - n_{S_D}$ to make inference for the generalization error. As such, only data in a *single* baseline period is utilized to create a decision rule. For all other periods, the full amount of data is utilized as the holdout data.

A simple way to think of it is that we have utilized a fraction of the data in the baseline period to create our model, and the rest of the data is utilized to obtain generalization errors for each period. If the distribution changes between periods, a structural break has occurred. Even though we need not rely on asymptotics, we once again outline that we are able to make valid inference for the generalization error in each period.

We define the expected generalization error associated with the distribution at time t, and the associated variance:

$$\mu_t \equiv \mathrm{E}[\mathcal{L}(S_D; i_t)] \tag{11}$$

$$\sigma_t^2 \equiv \mathrm{V}[\mathcal{L}(S_D; i_t)] \tag{12}$$

where i_t is a draw from the unknown distribution P_t.

Assuming that the regularity conditions hold and we have enough data to utilize asymptotic normality – as outlined in section 2.1 – we are able to make valid inference for all the periods

with the two estimators:

$$\hat{\mu}_t \equiv \frac{1}{|V_t|} \sum_{i \in V_t} \mathcal{L}(S_D; i) \tag{13}$$

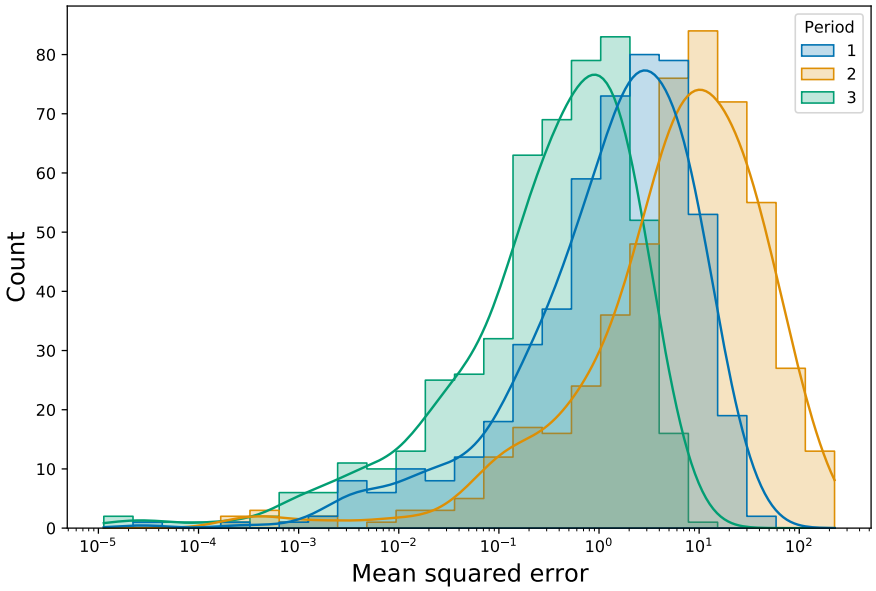
$$\hat{\sigma}_t^2 \equiv \frac{1}{|V_t| - 1} \sum_{i \in V_t} \left(\mathcal{L}(S_D; i) - \hat{\mu}_t \right)^2 = \frac{1}{|V_t| - 1} \sum_{i \in V_t} \left(\mathcal{L}(S_D; i) - \frac{1}{|V_t|} \sum_{i \in V_t} \mathcal{L}(S_D; i) \right)^2 \tag{14}$$

where V_t denotes the set of data available at time t which is *not* in S_D and $|V_t|$ denotes the amount of elements in the set. This corresponds to S_H for period 1 and all available observations for $t > 1$. As such, we are able to consistently estimate the mean of the generalization error in each period.

Another way to to think of this method is that we have T independent samples, one for each period, and if the DGP is unchanged, then the T different independent samples come from the same population, and thus will have the same characteristics. It is important to note that changes in *anything* in the DGP that influences the performance of the model can cause a break. This could be changed covariances between covariates, heteroscedasticity in the error term, changes in parameter values (both mean and for covariates) or changes in any other structure that this specific model utilizes to make predictions.

To illustrate this and display the simplicity of the procedure, we have created an example with a univariate model in three different periods. The covariate is standard normally distributed and has a baseline associated parameter value of 4 and a normally distributed error term with a mean of 0 and a standard deviation of 2. In period 1, we have 2000 observations, 75% of which are utilized to fit an OLS model (i.e. the decision rule $F(D)$ is obtained by fitting a regression to the development data) and 25% of which are utilized as a holdout data set. In period 2 and 3, we have 500 observations, utilized as holdout data sets. Here we exploit that the sample size in each period can vary to achieve equal sample sizes (for illustrative purposes), but in general one should except less observations in the baseline period due to the need for a development set. In period 2, we change the parameter value to 8, and in period 3 we change the standard deviation of the error to 1. The accuracy measure of choice is the mean squared error. For each holdout set, we compute predictions and calculate the mean squared error, and the result can be seen in figure 1.

Figure 1: Distribution of the mean squared error across periods



Note: The x-axis is logarithmically scaled

In this simple example, it can be visually be discerned that the distribution has changed, and thus the DGP has changed between periods. In real use cases, it will probably not be as clear, and we must use a statistical test to discern if there are any differences. As already established, the means, when scaled properly, asymptotically converge to standard normal distributions. However, it is not given that we must rely on asymptotics.

Each and every characteristic of the distributions should be the same, and as such the amount of statistical tests we can utilize is very large. We will make the case that equality of means is the most fitting characteristic to test, although certainly not the only one possible. This conclusion is drawn from two considerations: 1) There exists equality of means testing procedures that take into account when multiple comparisons are made (a topic we return to in section 2.3), and 2) it is common to focus on expected performance in the machine learning literature.

In regards to 1), there have been multiple procedures that have been developed and utilized as this is a common occurrence for researchers who are interested in effects between different treatments in experiments, see Maxwell and Delaney (2004) [11] for a thorough walk through, especially chapter 4. These tests obtain higher power than simply testing each pair with a given test and then utilizing the Bonferroni correction, while still controlling the type I error at the given significance level. In regards to 2), it seems to us that a break in the DGP which changes

characteristics other than the mean, but not the mean, are of lesser interest.

We note that we are interested in *which* of the periods differ, not just if periods differ, which rule out any tests which test equality for all periods at the same time without any way of detecting which periods differ in mean.

If $T = 2$, we propose to use a t-type test to test for equality between means. As we have no knowledge of how σ_i^2 will behave if the null is violated, we have two options: 1) Continue to require that our sample size in each period is large enough such that a central limit theorem applies and utilize tests with homogeneity and equal sample size assumptions [12], or 2) utilize tests that do not have homogeneity and equal sample size assumptions. In theory, a third possibility exists: Test whether the variances are equal before proceeding, but this will inflate the amount of type I errors by doing multiple tests [11][12] and is disregarded.

It has been shown that only a small loss of power occurs when using a heteroskedasticity and unequal sample size robust t-test (Welch's unequal variances t-test) when the assumption of homogeneity and equal sample sizes is met, but performing considerably better than Student's t-test when assumptions are not met [12][13]. We therefore recommend that one uses Welch's unequal variances t-test and do not require that the holdout sample sizes are sufficiently large for a central limit theorem to apply, hereby relaxing this assumption.

To quickly reiterate Welch's t-test, the test statistic for equality between the means of group i and j is calculated as:

$$t = \frac{\hat{\mu}_i - \hat{\mu}_j}{\sqrt{\frac{\hat{\sigma}_i^2}{|V_i|} + \frac{\hat{\sigma}_j^2}{|V_j|}}} \tag{15}$$

which is approximately t-distributed with ν degrees of freedom calculated from the Welch-Satterthwaite equation [12]:

$$\nu = \frac{\left(\frac{\hat{\sigma}_i^2}{|V_i|} + \frac{\hat{\sigma}_j^2}{|V_j|}\right)^2}{\frac{\left(\frac{\hat{\sigma}_i^2}{|V_i|}\right)^2}{|V_i|-1} + \frac{\left(\frac{\hat{\sigma}_j^2}{|V_j|}\right)^2}{|V_j|-1}} \tag{16}$$

As a general rule we have no a priori expection of which way the two means differ, and thus we recommend two-sided testing. This procedure is readily implemented in many programs, and in Python is implemented in *SciPy.stats.ttest_ind* with *equal_var = False* [14].

We now have outlined how to determine whether the sample mean of two different periods are

the same when $T = 2$. However, simply repeating this procedure when $T > 2$ will inflate the amount of type I errors, also known as the multiple testing problem.

2.3 The multiple testing problem

In this section we outline the multiple testing problem when comparing multiple means, and discuss some of the pros and cons of correcting. We go on to recommend the Games-Howell test, which corrects for multiple comparisons.

When repeatedly testing at a given significance level α, the probability of making a type I error, i.e. rejecting the null when it shouldn't be rejected, rises beyond the given significance level. As outlined in Abdi (2007) [15], the probability of making at least one type I error for a family of C independent tests at a significance level of α is given by:

$$1 - (1 - \alpha)^C \tag{17}$$

which is called the family-wise error rate (FWER). This number quickly rises to be quite large, i.e. for $C = 10$ the probability of making a type I error is 0.401.

Much has been written on whether to control the amount of type I errors per test or per family of tests, far beyond what can be encapsulated in this paper, and opinions differ on this matter [16]. Perhaps the most broadly accepted answer is 'it depends'. One reason not to control the FWER is that it increases the amount of type II errors (decreases the power of the test), i.e. not rejecting the null when the null does not hold [17]. How to handle this trade-off depends on the researchers goals, but it also highlights the need to consider which method is used for correcting, as some methods results in a higher loss of power than others. A common correction procedure is the Bonferroni correction, which is overly conservative and thus results in comparatively many type II errors [16], but can be used for any type of test with a significance level. This is part of the reason why we consider group means, as more powerful tests exist [18].

Another way of minimizing the problem is to reduce the amount of comparisons. In the case of detecting structural breaks, one might have a priori knowledge that could guide which comparisons are made. One could also only be interested in testing against the baseline (see e.g. Dunnett's test [19]). The most general case is when all pairwise comparisons must be made, and in this case we recommend to control the FWER. This stems from the fact that 1) the amount of comparisons quickly rises, 2) there exist mean tests where the loss of power is not as large as the Bonferroni correction, and 3) the null hypothesis of no breaks at all is of interest to us.

However, this is only a recommendation, and one could utilize multiple Welch's t-tests and not control.

We once again note that we have no knowledge of how the variance behaves under the alternative hypothesis, and therefore we consider tests which are robust to unequal sample sizes and or unequal variances. As mentioned earlier, Maxwell and Delaney (2004) has a thorough walk through in chapter 4 [11]. For multiple comparisons with unequal variances and or unequal sample sizes, they recommend either Dunnett's T3 [20] for small sample sizes (i.e., fewer than 50 per group) and the Games-Howell procedure for larger sample sizes [21] is recommended. The Games-Howell procedure is also found to have the highest power in another simulation study [18]. We proceed with the Games-Howell test as we seldom expect to have as few as 50 observations per holdout group.

The Games-Howell test [21] has the same assumptions (independence of observations and regularity conditions) and equations as Welch's t-test for test statistic and degrees of freedom:

$$t = \frac{\hat{\mu}_i - \hat{\mu}_j}{\sqrt{\frac{\hat{\sigma}_i^2}{|V_i|} + \frac{\hat{\sigma}_j^2}{|V_j|}}} \tag{18}$$

$$\nu = \frac{\left(\frac{\hat{\sigma}_i^2}{|V_i|} + \frac{\hat{\sigma}_j^2}{|V_j|}\right)^2}{\frac{\left(\frac{\hat{\sigma}_i^2}{|V_i|}\right)^2}{|V_i|-1} + \frac{\left(\frac{\hat{\sigma}_j^2}{|V_j|}\right)^2}{|V_j|-1}} \tag{19}$$

The null hypothesis that $\mu_i = \mu_j$ is then rejected if $|t| > \frac{q(\alpha,T,\nu)}{\sqrt{2}}$, where q is the studentized range distribution, α is the significance level and T is the amount of periods (groups). This test is not as readily implemented, but the studentized range distribution is implemented in the package *statsmodels* in *statsmodels.stats.libqsturng*, with p-values being able to be calculated from *psturng* [22].

Utilizing this procedure also allows for an arbitrary number of breakpoints and discerning whether periods separated by two breakpoints are the same, which could be useful in instances where the data oscillates between two DGP's, e.g. in a business cycle perspective.

3 Simulation

This section implements a simulated prediction problem and includes different setups to offer evaluation and proof-of-concept of the method. The methodology is introduced, and three setups with multiple periods are included to showcase the Games-Howell test, with the remainder of

the tests utilizing Welch's t-test to showcase the capabilities of the method and eliminate any influence from multiple testing corrections.

3.1 Structure

The simulation study follows the ADEMP structure outlined in Morris, White and Crowther (2019) [23], namely aims, data generating process, estimands and other targets, methods and performance measures.

3.1.1 Aims

The aim of this simulation study is mainly to offer a proof-of-concept that it works for prediction problems, both for multiple periods and for single pairwise comparisons. The power is estimated in a wide range of different setups.

3.1.2 Data generating process

The data generating process (DGP) utilized is draws from a known model, rather than repeated sampling from a given data set. This is chosen such that we are able to control and vary the breaks in the DGP. The factors in the DGP are varied mostly one-by-one to reduce running time, but if a priori we expect interaction effects two factors are varied together.

The specific DGP to generate n_t draws within each period T utilized are K 'primitive' random standard normal covariates. Covariances and standard deviations are random draws from a standard normal distribution (absolute value taken for standard deviations). From the K random normal covariates we generate both squared terms and interaction terms, hereby increasing the amount of covariates to $\binom{K}{2} + 2K$. This is done to ensure interdependencies within the data, as one would expect from real data. 75% of these covariates (rounded down) randomly have a parameter of zero. The remaining 25% (rounded up) have non-zero covariates that are standard normal draws. The target is generated as the sum of the product of the parameters and covariates plus a normally distributed error term with mean zero and variance σ_t.

The choice to draw from the normal distribution is to ensure that the regularity conditions are fulfilled. To ensure replicability, the seed it set once, and only once, in each notebook.

To reiterate the most fundamental information, with standard values in parenthesis if applicable:

- T, the amount of periods,

- n_t, the amount of observations in each period (50000),
- K, the amount of primitive covariates (10),
- σ_t^2, the standard deviation of the error (4),
- whether the same parameters are changed when repeated breaks occur,
- the development share in the baseline period (75%),
- the share of changed parameters (10%, corresponding to a single parameter for $K = 10$)

Any change in the DGP between two periods constitutes a break. We implement two breaks: Changing parameters and changing the variance of the error term. When changing parameters, we change a fraction of the non-zero parameters to a new draw from a standard normal distribution. If multiple breaks occur, it is both implemented such that 1) the same parameters are changed and 2) a new random subset of parameters are changed.

Many of the elements of the DGP are draws from a random normal distribution (covariances and standard deviation of covariates and parameters) and random selection of breaks. This large element of randomness is to ensure that the results are not an artifact due to a choice that we have made in these design steps.

3.1.3 Estimands and other targets

The target in this simulation is whether the null hypothesis is rejected, with focus on the power of the test and the amount of type I errors.

3.1.4 Method

The theory outlined could, as outlined, both be used for classification and prediction problems. A prediction task is chosen, for no particular reason. The method utilized to predict the target variable is the LASSO [24], which is a regularized regression model. This decision is made in tandem with our DGP decisions, as we expect that the LASSO will fare acceptably under the given circumstances (i.e. there are covariates with parameters of zero). Furthermore, the LASSO, and regularization in general, is widely known and utilized in the social sciences [25], hereby showcasing a use case with which many readers are familiar. The accuracy measure chosen is the mean squared error, as is customary for many models in prediction and regression problems.

The hyperparameter is chosen through 2 times repeated 5 fold cross validation with a development share of 75% in a given baseline period. Ideally 10 times repeated 10 fold cross validation would be preferred as this has been shown to have good replicability [26], but as most of the computing time is utilized here, the amount is reduced. 25 hyperparameters logarithmically distributed between 10^{-5} to 100 are iterated through. The best hyperparameter is chosen and a model trained on the whole development set is used to generate generalization errors for the holdout set in each period.

3.1.5 Performance measures

As we are interested in the power or amount of type I errors, the natural performance measure is the rejection rate, $\Pr(p_i < \alpha)$, which corresponds to either the amount of type I errors or the power of the test, dependent upon setup (i.e. if we have made a change between the periods being compared or not). Both the rejection rate mean estimator and Monte Carlo standard errors stem from Morris, White and Crowther (2019) [23]. The estimate of the rejection rate is the empirical mean:

$$\widehat{\Pr}\left(p_i < \alpha\right) = \frac{1}{n_{sim}} \sum_{i=1}^{n_{sim}} \mathbb{1}(p_i < \alpha) \tag{20}$$

Due to the inherent uncertainty when utilizing simulation methods, Monte Carlo standard errors are reported, calculated as:

$$\widehat{SE}\left[\widehat{\Pr}(p_i < \alpha)\right] = \sqrt{\frac{\frac{1}{n_{sim}} \sum_{i=1}^{n_{sim}} \mathbb{1}(p_i < \alpha) \cdot \left(1 - \left[\frac{1}{n_{sim}} \sum_{i=1}^{n_{sim}} \mathbb{1}(p_i < \alpha)\right]\right)}{n_{sim}}} \tag{21}$$

To decide upon n_{sim}, one could target a specific standard error. In an ideal case, we would also do this. To ensure that the running time of the program does not become too large trying to achieve specific standard errors and we do not have two methods we wish to compare against each other (thus creating a target standard error which results in a significant difference), we decide to do 250 repetitions.

3.2 Results

In this section we report the results for our simulations. First some use cases with multiple periods ($T = 6$ or $T = 12$) is reported. This is followed by a section where only a single comparison is made between two periods, hereby focusing on the factors varied in each comparison, removing any power losses due to correcting. All tests utilize a significance level of 5%. Due to space constraints more tables are available in the appendix.

3.2.1 Multiple comparisons

We first examine a baseline with 6 periods and no breaks. Rejection rate should be less than 5%, as this corresponds to the amount of type I errors, which we control at a significance level of 5% utilizing the Games-Howell procedure. The estimated rejection rate, seen in table 2, is 2.8% with a standard error of 1%. This indicates that the Games-Howell test may be conservative for this given setup, and controlling at less than 5%.

Table 1: No breaks, 6 periods

	0
Rejection %	0.028
Standard error	0.01

We now examine two setups of varying length ($T = 6$ and $T = 12$) with two breaks of 20%. Changed parameters are randomly selected (i.e. not necessarily the same parameters are changed). This is done to examine the loss of power associated with a higher amount of periods.

We denote periods with the same DGP as being within a 'paradigm', and we denote the paradigms consequently as 1, 2 and 3 (2 being 20% different from 1, and 3 being 20% different from 2). The models are trained on a baseline period from paradigm 1. As such, we have a setup with three paradigms of two periods and a setup with three paradigms of four periods.

Within paradigms rejection rates should be at most 5%, this corresponds to the amount of type I errors. Between paradigms, rejection rates should be as high as possible, as this corresponds to the power.

The results are reported in table 2 and in table 3. What is reported is whether *any* significant breaks have been found. Within thus compares all periods to all periods that come from the same paradigm, and between compares all periods to all periods not from the same paradigm. We further report where the significant differences are found, i.e. 1:1 reports the rejection rate within paradigm 1, and 1:2 reports the rejection rate between a period in paradigm 1 and period in paradigm 2.

We see that the amount of type I errors is very low (0.4% and 1.2%, respectively), again pointing

to the Games-Howell being conservative for the given setup. No clear pattern exists in where the type I errors are made.

The power also remains high, albeit lower for the higher amount of periods, as expected, with estimates of 96% and 95.2%, respectively, and no significant difference. Here we see a clear pattern in where the type II errors are made, with all occurring between paradigm 2 and 3. This is no surprise, as the model is trained on a period from paradigm 1. This is not the case for paradigm 2 and 3, and we therefore expect the generalization error to be distributed with larger variances, hereby reducing power.

Table 2: 2 breaks, 6 periods

	Within	1:1	2:2	3:3	Between	1:2	1:3	2:3
Rejection %	0.004	0.004	0.0	0.0	0.96	1.0	1.0	0.96
Standard error	0.004	0.004	0.0	0.0	0.012	0.0	0.0	0.012

Table 3: 2 breaks, 12 periods

	Within	1:1	2:2	3:3	Between	1:2	1:3	2:3
Rejection %	0.012	0.0	0.004	0.008	0.952	1.0	1.0	0.952
Standard error	0.007	0.0	0.004	0.006	0.014	0.0	0.0	0.014

We conclude that the method works as intended, controlling the amount of type I errors at (significantly) less than 5%, but still achieving high power.

3.2.2 Single comparison

The following results utilize Welch's t-test, hereby focusing solely on the capabilities of the method in a single instance where we compare two periods.

In table 4 we vary the variance of the error in all periods, and change a single parameter. The standard deviation is increased two fold five times. The power remains relatively unchanged with standard deviations of up to 16. The power then falls to 88.8%, 79.2% and 66.4%, respectively, but we also note that this is with quite high standard deviations of up to 128, compared to the covariates which are normally distributed with a standard deviation drawn from a standard normal.

Table 4: Different variances with break

	4	8	16	32	64	128
Rejection %	0.968	0.964	0.944	0.888	0.792	0.664
Standard error	0.011	0.012	0.015	0.02	0.026	0.03

In table 5 we vary the amount of observations in each period. This has two effects: We both reduce the amount of training data for a fixed development share of 75%, and we reduce the amount of IID samples in each period. The power remains high for as few as 500 observations in each period, but is increasing in amount of observations as expected. This showcases how this method can be utilized even when working with relatively small amounts of data.

Table 5: Amount of observations

	500	1000	2500	5000	10000	20000	40000
Rejection %	0.932	0.972	0.968	0.96	0.98	0.988	1.0
Standard error	0.016	0.01	0.011	0.012	0.009	0.007	0.0

As such, we end the testing of our method using simulations. We conclude that the method can be used across a wide variety of setups, with little to no loss of power, with the largest loss of power occurring when dealing with very noisy data.

4 Discussion

When evaluating this method, it is important to note what it can and what it cannot tell us, and thus, how one should interpret a significant difference in the distribution in the generalization error.

What the method can tell us is that *something* has changed between two periods, but not *what* has changed between two periods. As such, it could be an intercept that has changed, a changed parameter, a changed error distribution etc. For further work, if one is not interested in intercept changes, a heuristic where each period is mean centered before any analysis starts could be developed.

Furthermore, it is very important to note that we do not state that a model trained on *both*

periods, with information in regards to what period each observation is part of, would perform worse, e.g. a tree-based model could split on period at the first decision node, hereby effectively creating a different model for each period. If one is interested in sample delimitation in regards to now-casting, further work could be to develop a method where the accuracy of a model trained on k (with $k < T$) periods is compared to a model trained on $k + 1$ periods. Then an iterative procedure could be developed starting with $k = 1$, training a model on the last period as the baseline model, comparing to a model trained on the two last periods. If the performance of the model is not worse, repeat for $k = 2$ and iterate backwards until all periods are included or a significant decrease in performance is encountered. This comes back to the demarcation in the times series literature: Are we interested in detecting breaks or how to handle breaks when doing prediction or classification?

5 Conclusion

In this paper we propose a novel method to detect structural breaks in prediction and classification problems with repeated cross sections. The method utilizes holdout data to generate IID samples of generalization errors in each cross section, allowing us to redefine structural breaks in terms of changes in the distribution of the generalization error. No distributional assumptions are required, making the method widely applicable. It is shown that the mean of the distributions can be consistently estimated, but Welch's t-test and the Games-Howell procedure which do not rely on asymptotics are recommended, dependent upon whether the researcher corrects for multiple comparisons or not. The means testing procedure is very general, and allows for an arbitrary number of breakpoints and tells us whether paradigms separated by more than one breakpoint are the same. A prediction problem utilizing a LASSO predictor is simulated, showing that the method has high power across a wide variety of setups.

6 References

[1] G. C. Chow, "Tests of Equality Between Sets of Coefficients in Two Linear Regressions," *The Econometric Society*, vol. 28, no. 3, pp. 591–605, 1960.

[2] J. Bai and P. Perron, "Computation and analysis of multiple structural change models," *Journal of Applied Econometrics*, vol. 18, no. 1, pp. 1–22, 2003.

[3] P. Perron, Y. Yamamoto, and J. Zhou, "Testing jointly for structural changes in the error variance and coefficients of a linear regression model," *Quantitative Economics*, vol. 11, no. 3, pp. 1019–1057, 2020.

[4] M. H. Pesaran, D. Pettenuzzo, and A. Timmermann, "Forecasting time series subject to multiple structural breaks," *Review of Economic Studies*, vol. 73, no. 4, pp. 1057–1084, 2006.

[5] P. Perron and Y. Yamamoto, "Testing for Changes in Forecasting Performance," *Journal of Business and Economic Statistics*, vol. 39, no. 1, pp. 148–165, 2021.

[6] C. Nadeau and Y. Bengio, "Inference for the generalization error," *Machine Learning*, vol. 52, no. 3, pp. 239–281, 2003.

[7] Dietterich T.G., "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural computation*, vol. 10, no. 7, pp. 1895–1923, 1998.

[8] Y. Bengio and Y. Grandvalet, "No Unbiased Estimator of the Variance of K-Fold Cross-Validation Yoshua," *Journal of Machine Learning Research*, vol. 5, p. 1089–1105, 2004.

[9] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[10] C. R. Rao, "Linear Statistical Inference and Its Applications," *New York, John Wiley*, no. Second Edition, 1973.

[11] S. E. Maxwell and H. D. Delaney, *Designing Experiments and Analyzing Data*, 2nd ed., 1990.

[12] G. D. Ruxton, "The unequal variance t-test is an underused alternative to Student's t-test and the Mann-Whitney U test," *Behavioral Ecology*, vol. 17, no. 4, pp. 688–690, 2006.

[13] M. Delacre, D. Lakens, and C. Leys, "Why psychologists should by default use welch's t-Test instead of student's t-Test," *International Review of Social Psychology*, vol. 30, no. 1, pp. 92–101, 2017.

[14] SciPy.org, "scipy.stats.ttest_ind," 2021. [Online]. Available: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind.html

[15] H. Abdi, "The Bonferonni and Šidák Corrections for Multiple Comparisons," *Encyclopedia of Measurement and Statistics*, p. 103–107, 2007. [Online]. Available: http://knowledge.sagepub.com/view/statistics/SAGE.xml

[16] D. L. Streiner and G. R. Norman, "Correction for multiple testing: Is there a resolution?" *Chest*, vol. 140, no. 1, pp. 16–18, 2011. [Online]. Available: http://dx.doi.org/10.1378/chest.11-0523

[17] K. J. Rothman, "No adjustments are needed for multiple comparisons," *Epidemiology*, vol. 1, no. 1, pp. 43–46, 1990.

[18] D. C. Sauder and C. E. DeMars, "An Updated Recommendation for Multiple Comparisons," *Advances in Methods and Practices in Psychological Science*, vol. 2, no. 1, pp. 26–44, 2019.

[19] C. W. Dunnett, "A Multiple Comparison Procedure for Comparing Several Treatments with a Control," *Journal of the American Statistical Association*, vol. 50, no. 272, pp. 1096–1121, 1955.

[20] ——, "Pairwise multiple comparisons in the unequal variance case," *Journal of the American Statistical Association*, vol. 75, no. 372, pp. 796–800, 1980.

[21] P. A. Games and J. F. Howell, "Pairwise Multiple Comparison Procedures with Unequal N ' s and / or Variances : A Monte Carlo Study," *Journal of Educational Statistics*, vol. 1, no. 2, pp. 113–125, 1976.

[22] Statsmodels, "psturng," 2021. [Online]. Available: https://www.kite.com/python/docs/statsmodels.stats.libqsturng.qsturng_.psturng

[23] T. P. Morris, I. R. White, and M. J. Crowther, "Using simulation studies to evaluate statistical methods," *Statistics in Medicine*, vol. 38, no. 11, pp. 2074–2102, 2019.

[24] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[25] J. Grimmer, M. E. Roberts, and B. M. Stewart, "Machine Learning for Social Science: An Agnostic Approach," *Annual Review of Political Science*, vol. 24, no. 1, pp. 1–25, 2021.

[26] R. R. Bouckaert and E. Frank, "Evaluating the replicability of significance tests for comparing learning algorithms," *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, vol. 3056, pp. 3–12, 2004.

7 Appendix

In table 6, we vary the standard deviation of the error across periods, but change no parameters. The standard deviation in the baseline period is 4, and both standard deviation increases and reductions are tested. We observe no type II errors, and the amount of type I error is controlled at a level below α.

Table 6: Varying standard deviation

Standard deviation	2	3	4	5	6
Rejection %	1.0	1.0	0.04	1.0	1.0
Standard error	0.0	0.0	0.012	0.0	0.0

Table 7 varies the amount of primitive covariates K, but keeps the *amount* of changed parameters equal (only 1 changed parameter), i.e. the share of changed parameters fall as K increases. We observe no trend, with powers remaining stable around 98% to 99.6%.

Table 7: Varying amount of covariates

K	10	15	20	25	30
Rejection %	0.992	0.988	0.98	0.984	0.996
Standard error	0.006	0.007	0.009	0.008	0.004

Table 8 is a three period setup with two breaks, trained on period 1 and reports the comparison between period 2 and 3, i.e. not trained on any of the periods being compared. Parameters are randomly selected both times. We observe no clear trend, but it seems the power loss is highest along and near the diagonal.

Table 8: Two breaks, random parameters

	10%	25%	50%	75%	100%
10%	0.988	0.976	0.992	1.0	1.0
	(0.007)	(0.01)	(0.006)	(0.0)	(0.0)
25%	0.988	0.984	0.976	0.996	1.0
	(0.007)	(0.008)	(0.01)	(0.004)	(0.0)
50%	1.0	1.0	0.968	0.968	0.976
	(0.0)	(0.0)	(0.011)	(0.011)	(0.01)
75%	1.0	0.988	0.984	0.976	0.972
	(0.0)	(0.007)	(0.008)	(0.01)	(0.01)
100%	1.0	0.992	0.984	0.96	0.972
	(0.0)	(0.006)	(0.008)	(0.012)	(0.01)

Table 9 is the same setup as table 8, but is limited to changing the same parameters both times. The power is decreasing in the amount of changed parameters.

Table 9: Two breaks, same parameters

	10%	15%	20%	50%	75%	100%
Rejection %	0.988	0.992	0.972	0.972	0.96	0.968
Standard error	(0.007)	(0.006)	(0.01)	(0.01)	(0.012)	(0.011)

Table 10 is a two period setup which utilizes different shares of development data combined with varying standard deviations for the error term. Power seems relatively constant for different standard deviations, but is decreasing in the standard deviation.

Table 10: Varying development shares and error standard deviation

	4	8	16	32
25%	0.996	0.984	0.944	0.88
	(0.004)	(0.008)	(0.015)	(0.021)
50%	0.992	0.96	0.932	0.888
	(0.006)	(0.012)	(0.016)	(0.02)
75%	0.98	0.964	0.932	0.856
	(0.009)	(0.012)	(0.016)	(0.022)
90%	0.992	0.964	0.924	0.856
	(0.006)	(0.012)	(0.017)	(0.022)