**ONLINE TOOL-WEAR PREDICTION USING GAUSSIAN PROCESS REGRESSION WITH BLOCKWISE ANALYTICS**

Max Ferguson, Raunak Bhinge, Jinkyoo Park, Kincho Law

**Abstract**

This paper describes how the condition of a milling machine tool was predicted using vibration and audio signals measured near the milling machine blade. We demonstrate that the vibration and audio signals contain sufficient information to predict the milling tool condition. Sensor data is aggregated into blocks that correspond to the individual actions of the CNC milling machine. We demonstrate that this block-wise analysis technique allows advanced machine learning models to be applied at near real-time speed without sacrificing accuracy. The tool-condition model is shown to be accurate, especially when predicting the condition moderately worn tools.

Keywords – Machine learning, Gaussian Process Regression, Tool condition monitoring

**1. INTRODUCTION**

Modern wireless sensors are able to capture data at a high sample rate, allowing changes in system parameters to be detected in real-time .

Several researchers have demonstrated that performing data aggregation on a wireless sensor can dramatically reduce the power consumption and bandwidth requirements without significant loss of information (Diao, Ganesan, Mathur, & Shenoy, 2007; Krishnamachari, 2002). In this paper we demonstrate that the controller data from a CNC milling machine can be used to aggregate real-time sensor data, drastically reducing bandwidth requirements, whilst retaining important information about the operating condition of the machine. We demonstrate that the aggregated data contains sufficient information to predict the condition of the milling machine tool at near real-time.

Reliable tool-condition monitoring can provide a number of benefits for the manufacturing industry, like improved product quality and the prevention of tool breakage. With the increasing availability of low-cost sensors, it is possible to collect real-time vibration and audio data from critical locations inside automated manufacturing machines. Previous researchers have demonstrated that the condition of the machine tool is can be inferred from features of the vibration and audio time series. (Reference) identified that there is a correlation between the tool-condition and the kurtosis coefficient of the audio time-series and the condition of the tool. In this paper we identify a number of features that can be used to predict the condition of a CNC milling machine tool. These features are calculated on the device and transferred to a cloud database, via a wireless connection. A Gaussian Process (GP) model is trained to predict the tool condition based on these features. Once trained, the GP model is used to predict the tool-condition in near real-time.

When aggregating time series data, it is common to discretized the time domain into finite blocks. In methods such as short-time Fourier transform (STFT), the time series is frequency content of the signal is calculated over local sections of the signal. In practice, this usually involves dividing a longer time-series signal into shorter segments of equal length and then computing the Fourier transform separately on each shorter segment. Whilst methods such as STFT are an effective way to monitor the transient nature of the frequency content, they suffer from the averaging effect; if the frequency content of the signal changes significantly during the STFT window, then the resultant STFT will represent a weighted average of the former and latter signals. This is highly undesirable, as the resultant SFTT does not represent the actual state of the system during the operation of the machine. To overcome this issue, we use the controller data to dynamically align the sampling windows with the actions performed by the machine.

The paper is organized as follows; In section 2, we explain the experimental setup and wireless data collection hardware. In section 3, we describe how the data is aggregated on the wireless sensor, and transmitted to a cloud server. In section 4 we describe how features are extracted from the sensor data using the periodogram of each time series, and in section 5 we describe how a Gaussian process regression model is trained to predict the tool condition of the CNC machine using the featurized data. Section 6 describes the results, and the paper is concluded with a brief summary and discussion.

**2. EXPERIMENTAL DESIGN AND MONITORING HARDWARE**

In this section we describe the experimental setup for collecting training and testing data. A number of simple parts were produced with a Computer Numerical Control (CNC) milling machine. As the parts were produced the condition of the milling machine blade deteriorated. The acceleration and acoustic signals inside the milling machine were measured throughout the duration of the milling process.

**2.1 Experimental Setup**

Figure . A Mori Seiki NVD1500DCG milling machine (left) and an InfiniteUptime sensor (right)

A Mori Seiki NVD1500DCG milling machine, similar to the one shown in Figure 1a, was programmed to produce a number of simple ‘parts’ by removing material from a solid steel block. Each part consisted of 20 separate cutting actions performed by the milling machine. On average, each cutting action was performed in about 3 seconds, and each part took about 1 minute to produce. The machine was instructed to produce parts until the cutting tool became severely damaged, or the cutting tool broke. A total of 23 tools were used to produce 100 parts.

The operating parameters of the machine were adjusted to artificially increase the rate of tool-wear. In a normal manufacturing environment, machine tools tend to last several days. In this experiment the operating lifetime of the machine tool was reduced to about 10 minutes by increasing the feed rate and the reducing the rotation speed.

An InfiniteUptime sensor was used to measure the audio and acceleration signal inside the milling machine while the machine was operational. The acceleration signal was recorded at 1000 Hz while the audio signal was recorded at 8000 Hz. The acceleration was measured in three axes. The sensor was waterproof, allowing it to be placed directly beneath the part being manufactured.

**3. DATA AGGREGATION**

**Recording the time series**

The acceleration and acoustic signals are measured continuously during the operation of the machine. We denote the three acceleration time-series signals as and the audio signal as . The acceleration signals are recorded with a 1000 Hz sample rate and the audio signals are recorded with an 8000 Hz sample rate. Initial investigation revealed that the direction of the measured acceleration is not a useful factor in prediction of tool condition, so the acceleration time series signals are combined:

Where is the magnitude of acceleration measured at the device. We use the symbol to refer to the time series signals collectively.

**Defining actions and action types**

The milling machine performs a number of operations to produce a part. We classify each operation as either a climb cutting, conventional cutting or air cutting operation, based on the type of cutting strategy that the machine is employing whilst performing an operation on the part. A physical interpretation of these actions has been described in an earlier work (Park J, 2015). We use the superscript to denote operation performed by the machine, and the subscript to denote the type of action (air cutting, conventional cutting, climb cutting). The time type of action being performed by the machine at any time, , can be obtained from the G-code provided by the milling machine [Raunak to check]. Figure 2 demonstrates that the production of a part in this work required 7 climb cutting actions, 8 conventional cutting actions. Each cutting action is separated by a brief air-cutting action, in which the machine pauses briefly between actions.

**Dividing the time series**

The milling machine G-code is used to divide the signal into a series of shorter signals , corresponding to the signal produced during operation with operation type *j*. The vibration signals contained 6142 points on average and the audio signals contained 49136 points on average.

Features are drawn from the Power Spectral Density (PSD) of the signal. Any change in the frequency content of the signal over time is assumed to be irrelevant to the prediction of tool condition. Welch’s method (Welch, 1967) is used with a Hann window to estimate the PSD for the signal. The time series signal is divided into successive blocks, with each block being defined by:

Where is defined at the window hop size and denotes the number of available frames. The Hann window is defined as follows:

Where is the length of the window. The periodogram of the block is calculated using the fast Fourier transform:

The Welch estimate of the power spectral density is given by:

The benefits of using Welches method over the Discrete Fourier transform are two-fold. The number of points in the resulting power spectrum is reduced and the random noise in the power spectrum is also reduced. A Hamming window length of 512 points was chosen for the vibration data and a 2048-point window was chosen for the audio signal. The number of points in each power spectrum is equal to the number of points in the window. Therefore, the transformation from the time domain to the frequency domain represents a dimensional reduction of approximately 12 times in the acceleration signal and 24 times in the audio signal.

Figure showing

1. Time series
2. Labelled time series
3. PSD

Maybe Ranauk wants to comment about transmitting the PSD.

**4 FEATURIZATION**

The aim of this section is to identify a set of features, which are correlated with the condition of the milling machine tool. The first four features are derived from the Power Spectral Density (PSD) defined in Section 3. As the tool condition deteriorates the periodogram changes. Several similarity measures are used to quantify the change in the periodogram, including the Euclidian distance and the Fréchet distance. In each case the periodogram, approximated by , is compared to a reference periodogram, . The reference periodogram represents the signal produced by a new tool. In this particular experiment the reference periodogram was obtained by averaging the first three periodograms for each action:

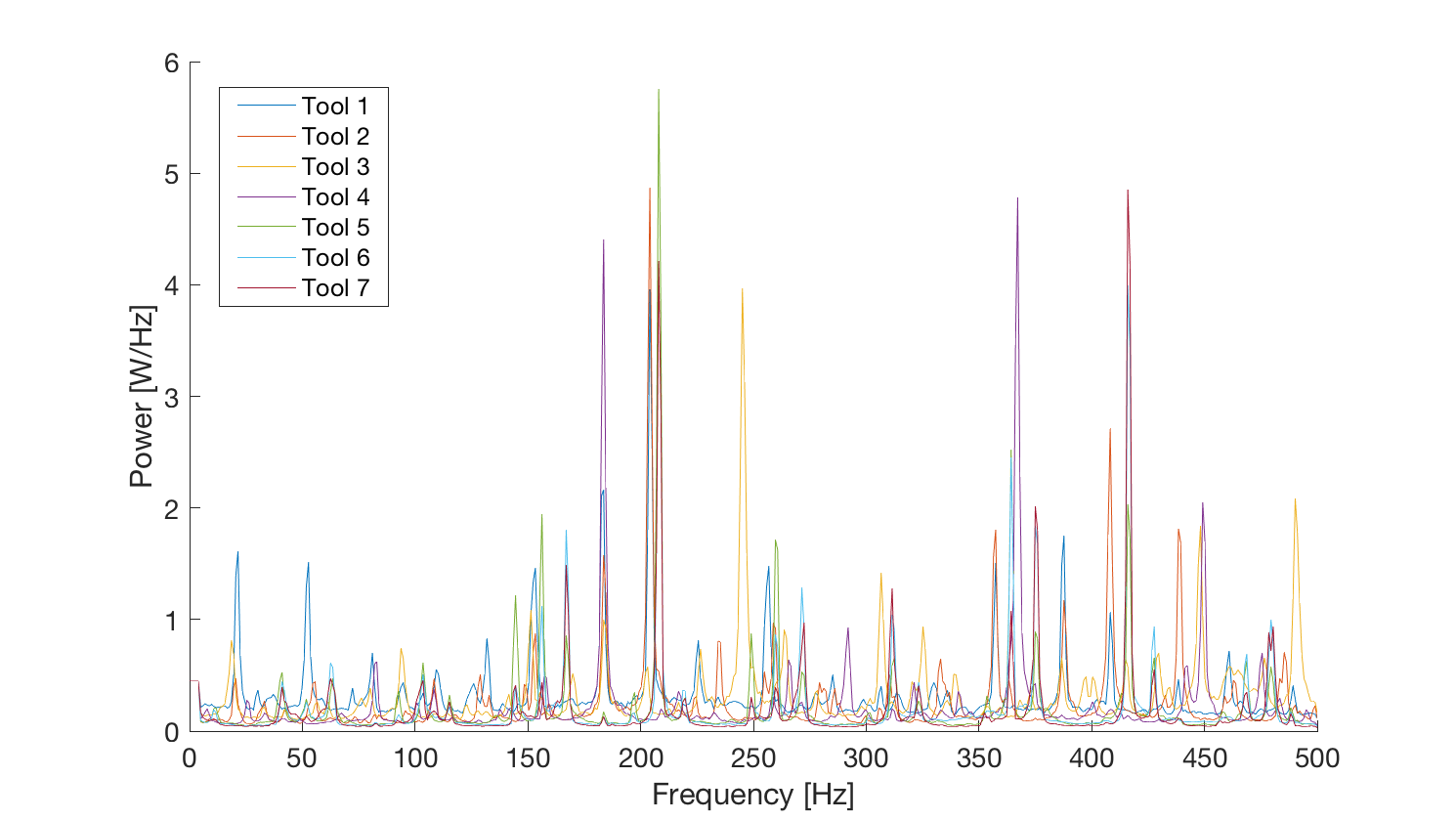


Figure 2. Reference periodograms for acceleration time series, conventional cutting. This figure will need to be replaced with a black/white one for publishing

**4.1 Signal Power**

The signal power can be approximated by integrating the periodogram with respect to frequency:

Where is the signal energy for the continuous signal In reality, the signal is expressed as a series of discrete points. Therefore, signal power can only be approximated from the periodogram:

The increase in signal power, between the reference signal and signal can be written as:

Where is the increase in signal power.

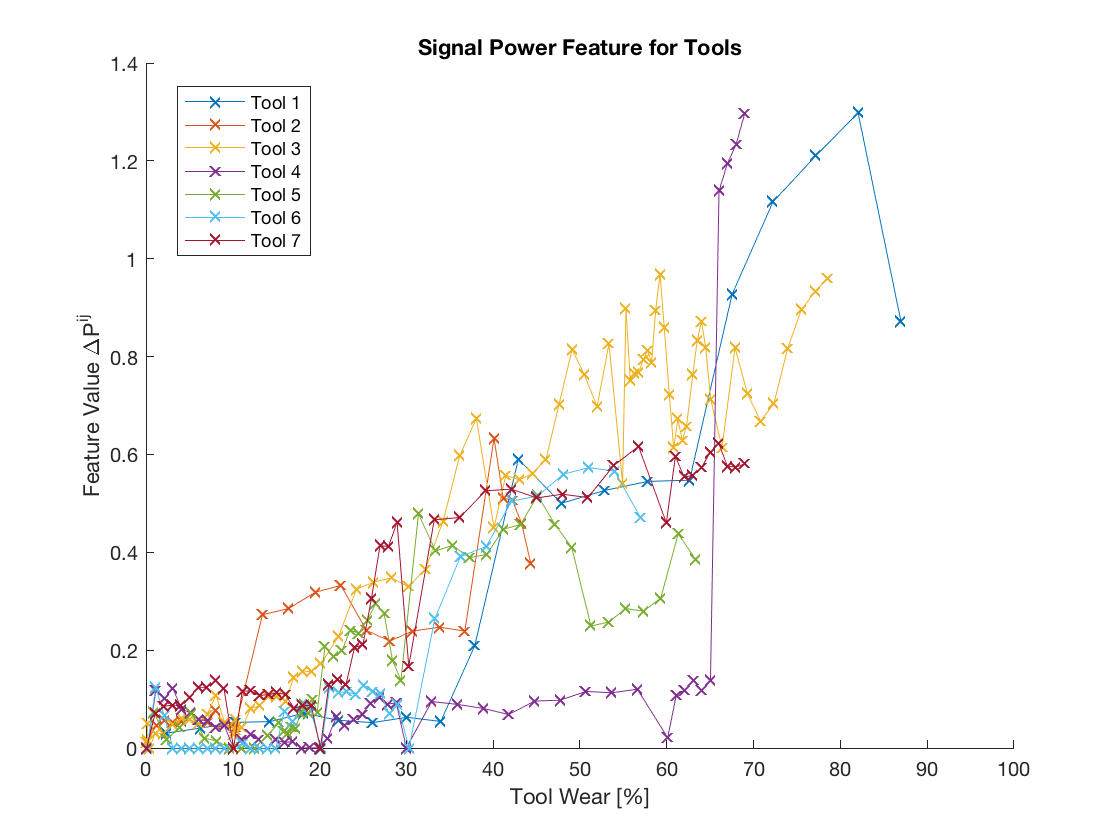


Figure . The signal power feature, , plotted against the amount of tool wear, for the conventional cutting operations in the training data set

**4.2 Magnitude of the transform function**

In defining this feature we assume that the periodogram is related to the reference periodogram by some transform function,

In discrete form we can write this as:

The value of the transformation function at frequency k is given by:

Summing the transformation function over the frequency domain:

Where is a feature representing the magnitude of the transform function.

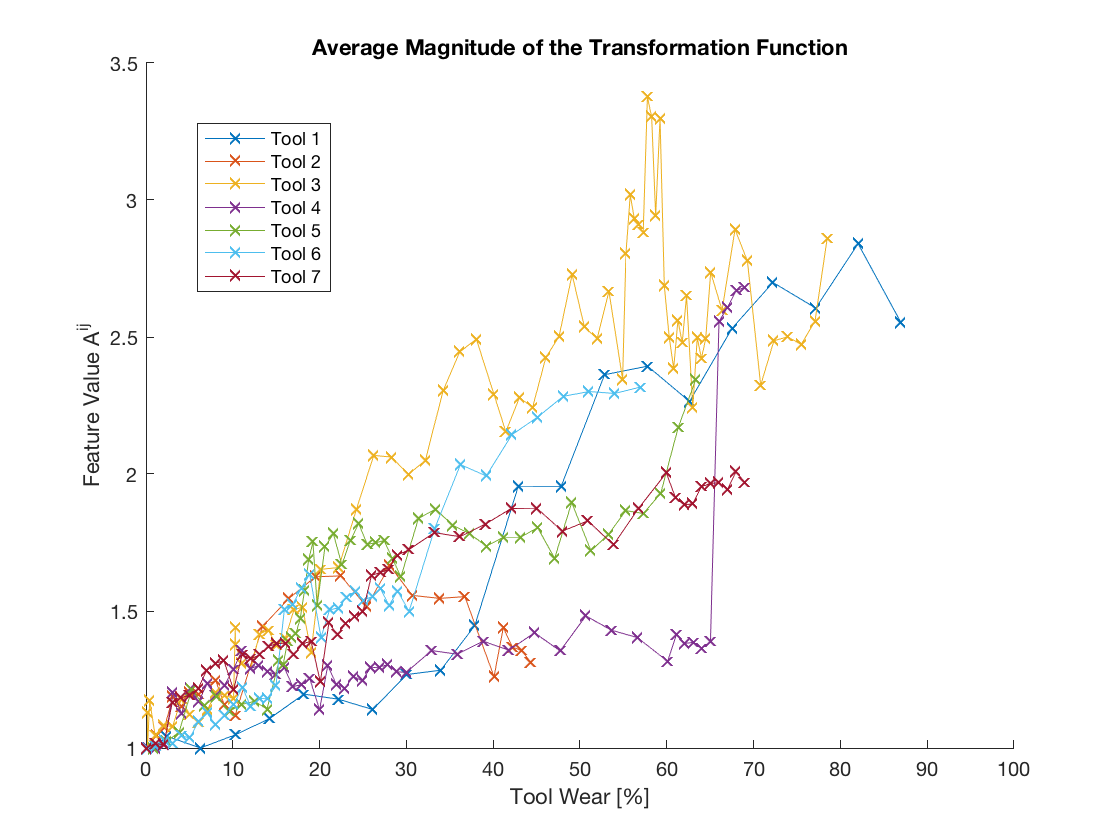
****

Figure . The transformation magnitude feature, , plotted against the amount of tool wear, for the conventional cutting operations in the training data set

**4.3 Fréchet** **distance between periodograms**

In defining this feature we treat the spectrogram as a curve in two-dimensional Euclidian space, with frequency as the first dimension and amplitude as the second. We define a new feature , as the Fréchet distance between the spectrogram and the reference spectrogram . The Fréchet distance is a measure of the similarity of two curves that accounts for both the location and ordering of points along the curve. An intuitive definition of the Fréchet distance is the minimum length of the leash required to connect a dog and its owner as they walk along two curves without backtracking.

The formal definition of the Fréchet distance between the two continuous curves P and Q is as follows. Let be the set of all continuous non-decreasing functions from onto . The continuous Fréchet distance between curves P and Q with and is given by:

where denotes the Euclidian distance. We define a new feature as the discrete Fréchet distance between the two curves:

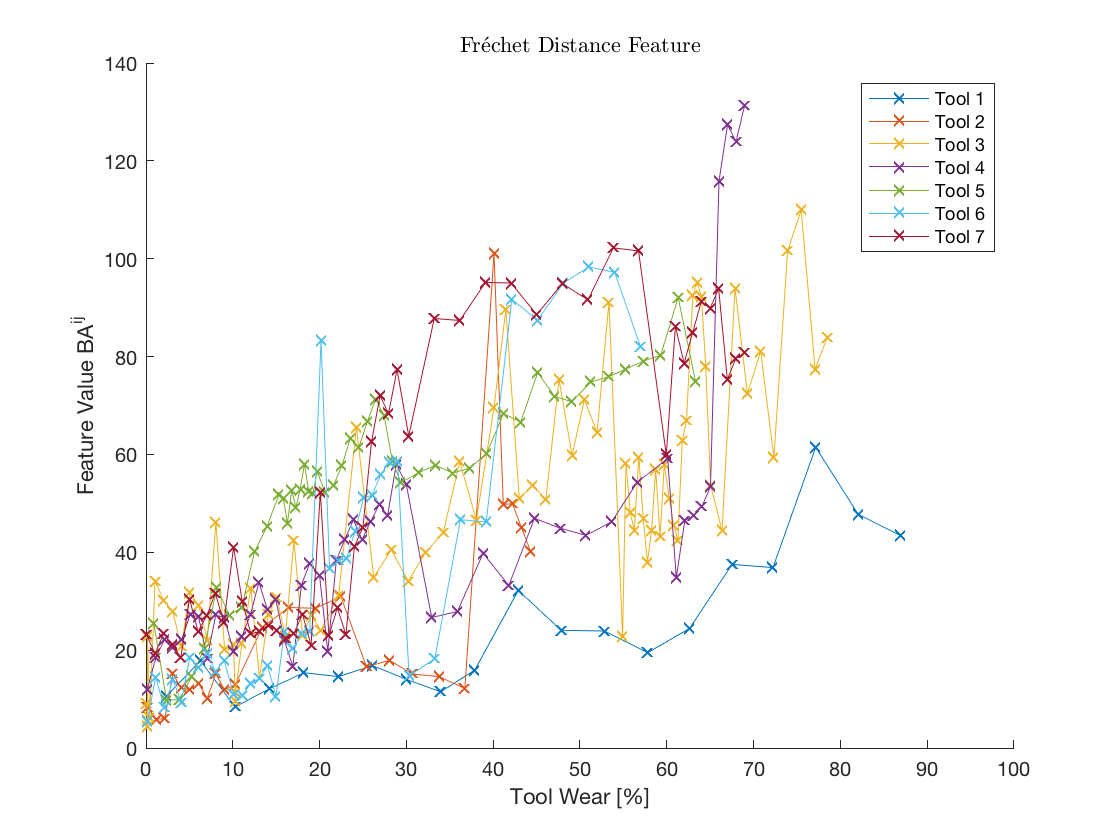


Figure . The Fréchet distance feature**,** plotted against the amount of tool wear, for the conventional cutting operations in the training data set

**4.4 Kurtosis coefficient**

Several authors have indicated that the condition of the tool is related to the kurtosis coefficient of the acoustic signal (ref). The kurtosis coefficient provides a way to identify sudden changes in the time series signal (ref). The kurtosis coefficient is calculated on the device at regular intervals *i:*

Where the signal mean and is defined as:

The kurtosis coefficient is also calculated for the reference time-series signal,

A new feature, is defined as the ratio of the kurtosis coefficient of signal , normalized by the kurtosis coefficient of the the reference signal

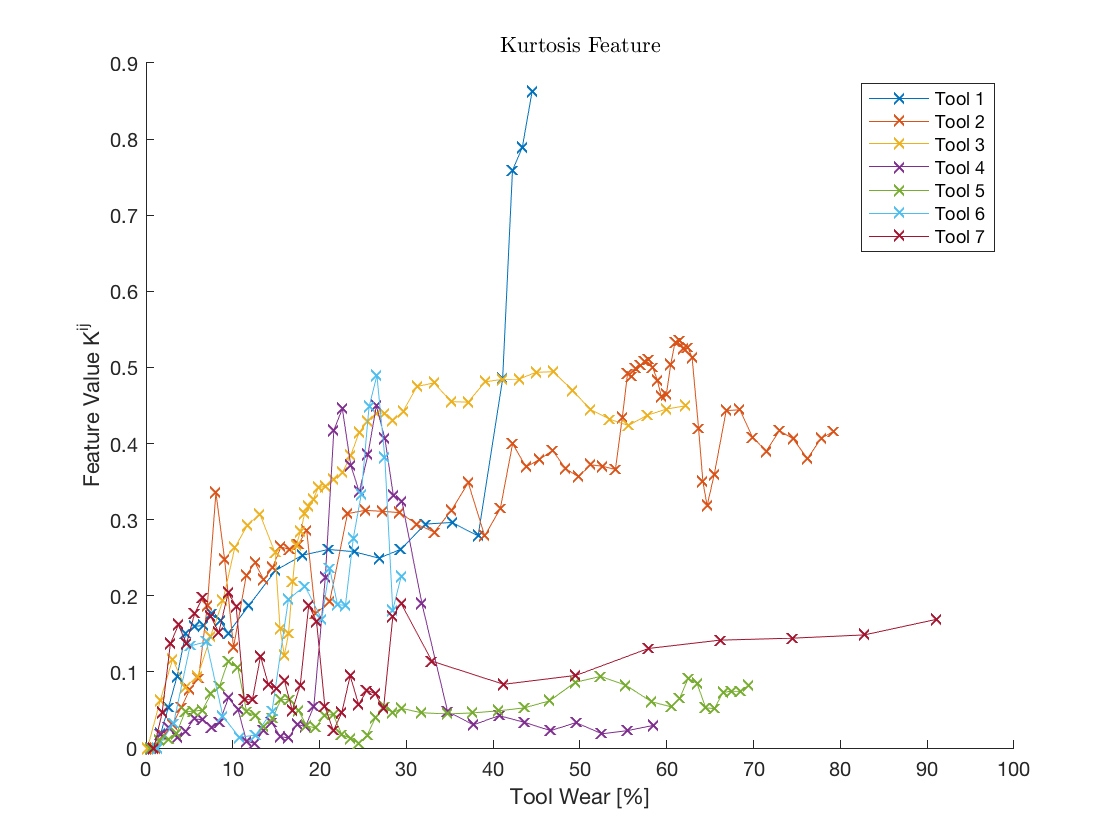


Figure . The kurtosis feature, , plotted against the amount of tool wear, for the conventional cutting operations in the training data set

**5. DEVELOPING A GAUSIAN PROCESS MODEL TO PREDICT TOOL CONDITION**

This section describes how the tool condition models are constructed using a Gaussian Process (GP). First, the fundamental concepts of GP regression are introduced. We then discuss how the features described in section 3 are used to build a construct a model to predict tool condition, for each machine operation. Lastly we describe how prediction models for each machining operation are aggregated to estimate the condition of the tool at any given time.

**5.1 Gaussian Process Regression**

Gaussian process regression (GPR) is supervised machine-learning method that performs particularly well with noisy data. The aim is to learn a mapping from the inputs to the target value. In general, we denote the input as and the target value as .InGPR, a Gaussian process (GP) is used as a prior to describe the distribution on the target function . A GP is a generalization of the Gaussian probability distribution  for which any finite [linear combination](https://en.wikipedia.org/wiki/Linear_combination) of [samples](https://en.wikipedia.org/wiki/Sampling_(statistics)) has a [joint Gaussian distribution](https://en.wikipedia.org/wiki/Multivariate_normal_distribution) (Rasmussen & Williams, 2006), As the GP is a multivariate Gaussian distribution over the function , it can be fully specified by its mean function and covariance function In a GP, and are not constant parameters but functions incorporating prior knowledge about the unknown function (Rasmussen & Williams, 2006).

|  |  |
| --- | --- |
|  | (1) |

The mean function captures the overall trend in the target function value, and is used to approximate the covariance by representing the similarity between the data points. In GPR, is often chosen to be a zero function (Radford, 1998). In this case the target function is fully described by the covariance kernel function The type of kernel function used to build a GPR model can strongly impact the accuracy of the prediction model. In this study the Automatic Relevance Determination (ARD) squared exponential covariance function was chosen:

|  |  |
| --- | --- |
|  | (2) |

where the kernel function is described by the hyper-parameters, and . The signal variance hyper-parameter quantifies the overall magnitude of the covariance value. The hyper-parameter vector   
is used to quantify the relevancy of the input features in when predicting the response .

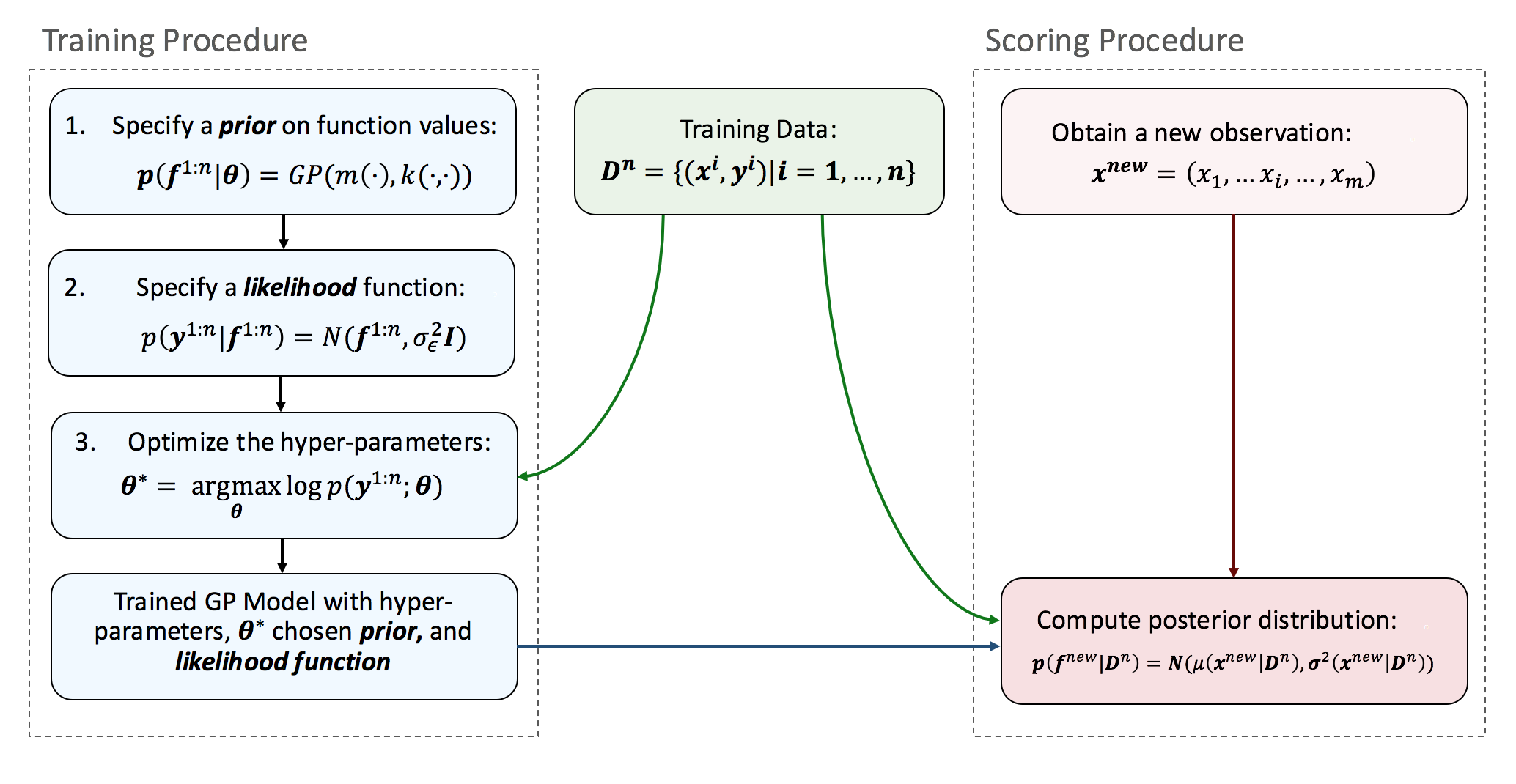


Figure . Flowchart showing GPR training and scoring procedure

**5.2 Tool Condition Model**

In this section we use GPR to develop a model to predict the condition of a milling machine tool , whilst the machine is performing some action *q.* We attempt to learn the unknown function by incorporating prior knowledge captured in historical data. Suppose the current data set is denoted by . The measured output corresponding to the new inputs and the historical outputs in the training data set, follow a multivariate Gaussian distribution:

|  |  |
| --- | --- |
|  | (3) |

where is the covariance kernel matrix defined as . The type of kernel function used to build a GPR model can strongly impact the accuracy of the prediction model. In this study the Automatic Relevance Determination (ARD) squared exponential covariance function was chosen [This is repeated but I prefer it here. I will remove it from the section 5.1 soon]:

|  |  |
| --- | --- |
|  | (4) |

where the kernel function is described by the hyper-parameters, and . The signal variance hyper-parameter quantifies the overall magnitude of the covariance value. The hyper-parameter vector   
is used to quantify the relevancy of the input features in when predicting the response .

The input features were chosen from the features described in section. It was decided that all of the features described in section 4 were beneficial to the model. More specifically, removing any of the features reduced the 5-fold cross validation accuracy. The input feature vector is defined using the measures from section 4:

**5.3 Noise Model**

GPR can also be used to account for conditions where there is noise in the observed outputs. Each observed value is assumed to contain some random noise , such that . We will assume that this noise follows an independent, identically distributed Gaussian distribution with zero mean and variance

|  |  |
| --- | --- |
|  | (5) |

It follows from the independence assumption that the noise model can be represented by adding a noise term to the kernel function:

|  |  |
| --- | --- |
|  | (6) |

Where represents Kronecker delta function which serves to selectively add the noise variance to the covariance value.

**5.4 Hyperparameter optimization**

The hyper-parameters are chosen to maximize the marginal likelihood of observations in a given training data set training data **.**

|  |  |
| --- | --- |
|  | (7) |

After optimizing the hyper-parameters the length-scale hyper-parameter vector can be used to determine the relevance of each feature, . A small value of indicates high relevancy of while a large value indicates low relevancy.

**5.5 Predicting tool condition**

Since the distribution conditional on any subset of the data is assumed to be Gaussian distributed, the posterior distribution on can be expressed as a Gaussian distribution (Rasmussen & Williams, 2006):

|  |  |
| --- | --- |
|  | (8) |

The posterior distribution on the tool condition can be expressed by its mean and variance both of which have closed-form representations:

|  |  |
| --- | --- |
|  | (9) |
|  | (10) |

**5.6 Smoothing Predicted Tool Condition**

As each prediction is based on a single milling machine action, the predictions are likely to contain a large amount of random noise. By incorporating the prior knowledge that the state of the tool is likely correlated with the previous tool condition we can smooth the predicted values. We define the correlation function between two successive predictions as:

We assume that any two predictions and are Then the expected value of the tool condition can be predicted:

**5. RESULTS**

It was hypothesized that the condition of the tool could be predicted from the acoustic and acceleration signal produced by the milling machine. The frequency content of the acceleration signal (ref), as well as the spectral energy density of the both the vibration and acceleration signals. This section describes the relevance of each feature to the tool condition model, as well as the overall performance of the model.

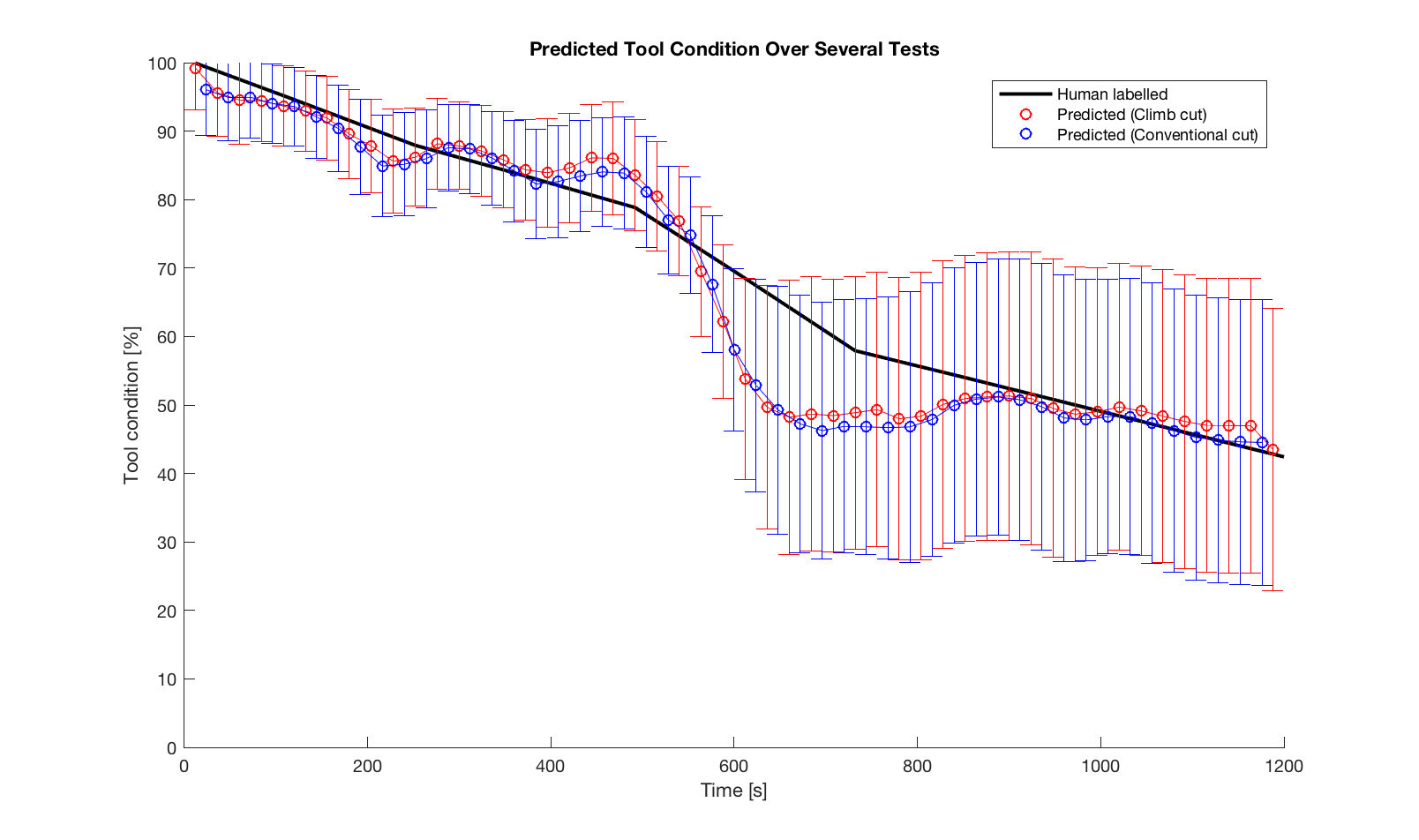


Figure . Predicted and measured tool condition plotted against time. The error bars show the standard deviation in the predictions. The climb cut and conventional cut predictions were made by separate GP models

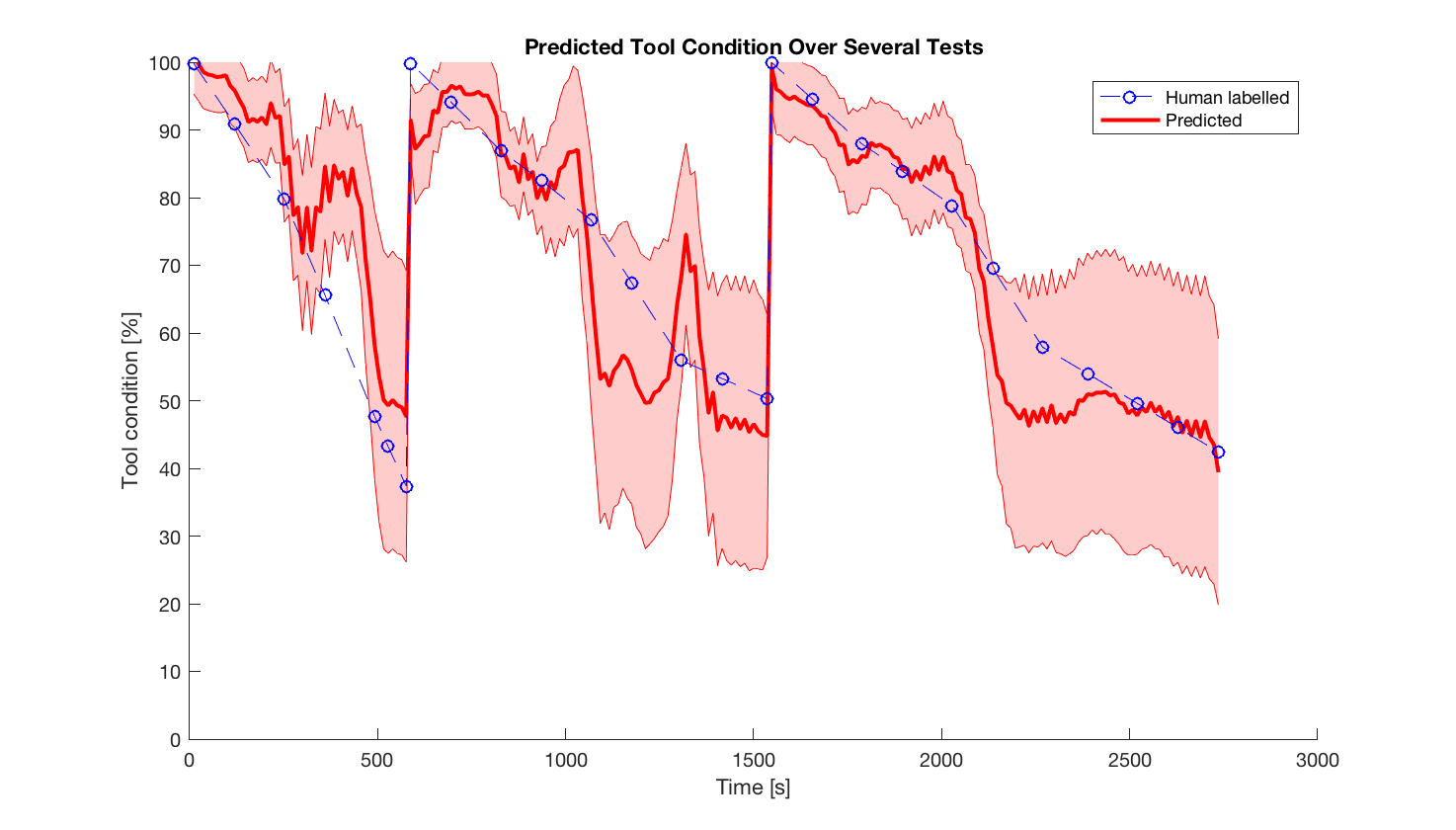


Figure . Predicted and measured tool condition for the entire testing set.

**6. DISCUSSION**

This study demonstrates how information from a milling machine controller can be combined with sensor data to predict the condition of the milling machine tool. The milling machine controller provided the information required to aggregate the time-series data over a set of finite intervals. The use of a non-parametric regression model, namely Gaussian Process Regression (GPR), allowed the complex relationship between the input features and target value to be modelled.

The predicted tool condition values were reasonably close to the human-labelled estimates, especially given the difficulties with quantifying tool condition. One common source of error is that the model is allowed to predict that the condition of the tool is increasing, due to the fact that each prediction is discrete and independent. This prediction

The GP model provides confidence bounds for the predictive estimations, which are useful when interpreting the reliability of a prediction at some arbitrary time. The confidence bounds would likely prove useful in a practical application where the tool-condition predictions were used to determine when to change machine tools.

The use of Welches method to calculate the power spectrum results in a noticeably smoother spectrum, with much less noise between the frequency peaks. Welches method also provides a way to reduce the time-series from an arbitrary length to specific number of points. If the power spectrum is calculated on the sensor using Welches method, the bandwidth requirements for transmitting the data would be significantly lower.

**Shortcomings**

The acceleration periodogram for a given machine action is highly dependent on a number of factors, such as the location and orientation of the acceleration sensor. While numerous measures were taken to control these additional factors it was evident that the reference periodograms were different for each part that was manufactured. This indicates that there are some unknown factors influencing the acceleration and audio signals.

Figure 8 shows that the prediction standard deviation tends to get larger as the predicted tool condition decreases. This is predominantly because the amplitudes of the acceleration signal generated by heavily worn tools have a much larger variance. For example, if a single flute on the tool is worn heavily the tool is no longer symmetric so corresponding acceleration signal will have a very high amplitude. However, if the tool is worn evenly the corresponding acceleration signal may have a much smaller amplitude.

**7. FUTURE WORK**

Many of the input features in this study were generated by comparing periodograms with a range of distance measures. These input features were then compared using the ARDSquaredExponetial kernel to obtain the Kernel covariance matrix. It is proposed that the periodograms from each machine action can be used as feature vectors directly. An appropriate covariance kernel would be chosen to compare periodograms, and regularization would be used to prevent overfitting. This method avoids the loss of useful information during the featurization step, at the expense of solving a more difficult optimization equation.

Modern learning methods such as Recurrent Neural Nets (RNN) often use predictions from previous time steps as input features when predicting the current state (Zhang & Chan, 2000). A similar approach could be used to extend the method described in this study. The predicted tool condition from the previous time step would be used as a feature when predicting the current tool condition. Incorporating previous predictions in this way may to reduce the random noise in the output values, and remove the necessity of post-prediction smoothing.

For this method to have practical applications it must generalize well to a range of different machines and machine operations. The GPR learning algorithm can be extended to a real-time adaptive learning algorithm. This would reduce data storage requirements and improve generalizability across different parameter spaces.

**7. REFERENCES**

1. R. M., Neal. (1996). *Bayesian learning for neural networks*. Springer-Verlag, New York.
2. R. Bhinge, J. Park, N. Biswas, M. Helu, D. Dornfeld, K. Law and S. Rachuri, “An Intelligent Machine Monitoring System Using Gaussian Process Regression for Energy Prediction,” *in Proceeding of IEEE* *International Conference on Big* *Data* (IEEE BigData 2014), 2014, Washington, DC.
3. J. Park, R. Bhinge, N. Biswas, M. Srinivasan, M. Helu, S. Rachuri, D. Dornfeld and K. Law, “A generalized data-driven energy prediction model with uncertainty for a milling machine tool using Gaussian Process,” *in Proceeding of ASME 2015 International Manufacturing Science and Engineering* Conference, 2015, Charlotte, NC.
4. Teramura, K. Okuma H. Taniguchi Y, Makimoto, S. Maeda S. “Gaussian Process Regression for Rendering Music Performance”, *in Proceeding of the 10th International Conference on Music Perception and Cognition,* 2015Sapporo, Japan.
5. V. Tresp, “A Bayesian committee machine,” *Neural Computation,* 12(11), 2000, pp. 2719–2741.
6. D. Nguyen-tuong, and J. Peters, “Local Gaussian process regression for real time online model learning and control,” *Advances in Neural Information Processing Systems*, MIT Press, 2008.
7. J. Shi, R. Murray-Smith, and D. Titterington, “Hierarchical Gaussian process mixtures for regression,” *Statistics and Computing*, 15, 2005, pp. 31-41.
8. A. Ranganathan, M. Yang, and J. Ho, “Online sparse Gaussian Process regression and its applications,” *IEEE Transactions on Image Processing*, 20(2), 2011, pp. 391-404
9. H. Xiao, and C. Echert, “Lazy Gaussian Process Committee for Real-Time online regression,” *in Proceeding of the Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI-13)*, Bellevue, WA, USA, 2013.
10. J. Quiñonero-Candela and C. Rasmussen, “A unifying view of sparse approximate Gaussian Process regression,” Journal of Machine Learning Reseach, 6, 2005, pp. 1939-1959.