Chapter 7

Spike Rate and Network Properties

After getting and studying the structure of reconstructed networks, we have revealed the relationship between the network structure and spiking dynamics. Spiking activity of nodes is obtained directly by applying the PTSD algorithm (see Appendix A) to the MEA recordings. In this chapter, we would like to study the dependence of spike rate of nodes with a network structure, gaining distinguishable network features in division groups of nodes, including particularly the excitatory incoming coupling strength and outgoing coupling strength (especially in positive outgoing coupling strength). Moreover, all network features can be employed to predict the spike rate of nodes in the study.

7.1 Dependence of spike rate

Since the spike rate is the number of spikes in a fixed time, which reflects the active level of nodes. It is believed that the spike rate is related to network features of nodes. We divide the spike rate into 10 decile groups depending on

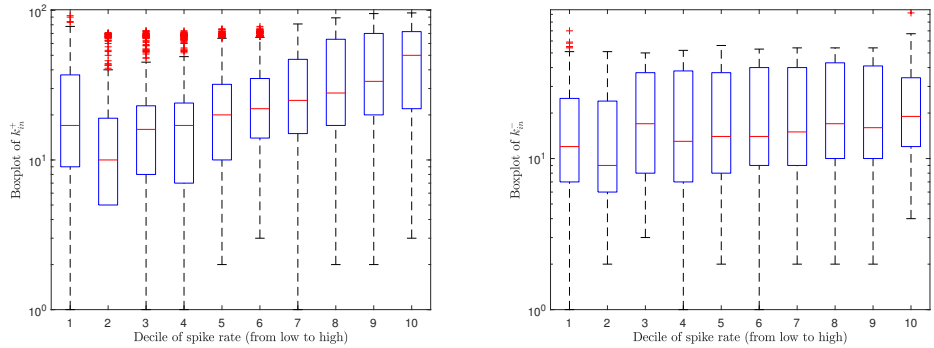
the spike rate–each group has about 410 nodes. Figure 7.1 shows the box plots of each group from a low spike rate to a high spike rate using MATLAB function "boxplot". The group "1" means 10% nodes with the lowest spike rate while group "10" means 10% nodes with the highest spike rate. It is clear that g_{in}^{+} and g_{out} are strongly related to spike rate, followed by k_{in}^{+} and k_{out}. Nodes with a high spike rate tend to have high structure features in these cases. In box plots of k_{in}^{+} and k_{out}, the same trend is observed. But the interquartile range is too large for k_{in}^{+} and there are many outliers for k_{out}. Box plots of k_{in}^{-} and g_{in}^{-} have a weak trend.

We plot the conditional probability density function of network features $k_{in}^{+}, k_{in}^{-}, k_{out}, g_{in}^{+}, g_{in}^{-}, g_{out}$ with nodes that have a spike rate higher or equal and lower than the median spike rate (denoted as HSR or LSR) for concrete comparison in Figure 7.2 7.3 7.4 7.5 7.6 7.7 7.8. 4 kinds of coupling strength are in logarithm forms with positive values. Distributions of $\ln|g_{out}|$ are in positive part and the negative part depending on the excitatory or inhibitory nodes.

Only excitatory incoming coupling strength (g_{in}^{+}, Figure 7.5) and outgoing positive coupling strength (g_{out} for $g_{out} > 0$, Figure 7.7) have a clear division for HSR and LSR nodes, which can be used to predict a node belongs to high spike rate or not depending on a threshold set for network features.

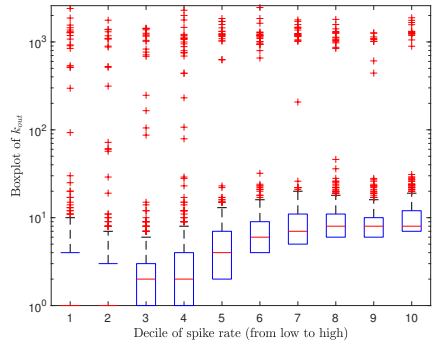
7.2 Prediction of high spike rate nodes

Some network features like g_{in}^{+} and g_{out} can be used to predict whether a node belongs to HSR or LSR since the spike rate has a strong correlation with these network features. We try to make use of g_{in}^{+}, g_{out} or all 6 network features separately to do such predictions.

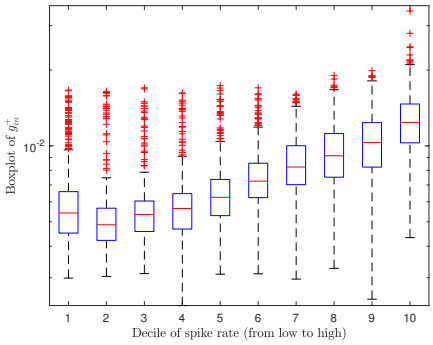


(a) k_{in}^{+}

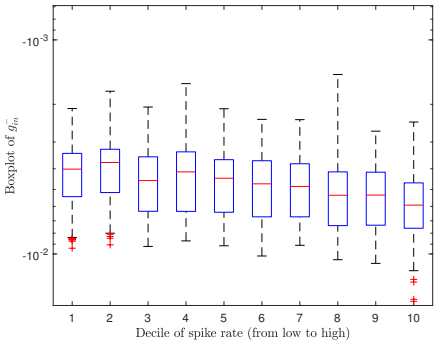
(b) k_{in}^{-}



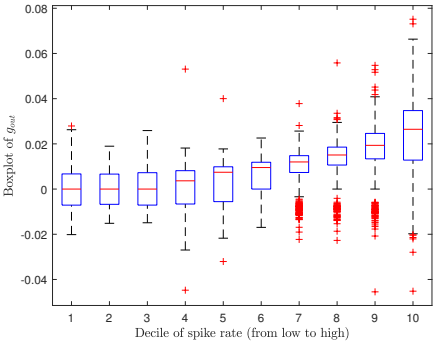
(c) k_{out}



(d) g_{in}^{+}

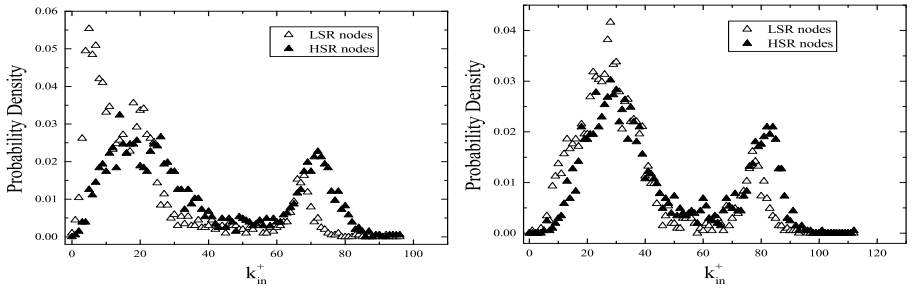


(e) g_{in}^{-}



(f) g_{out}

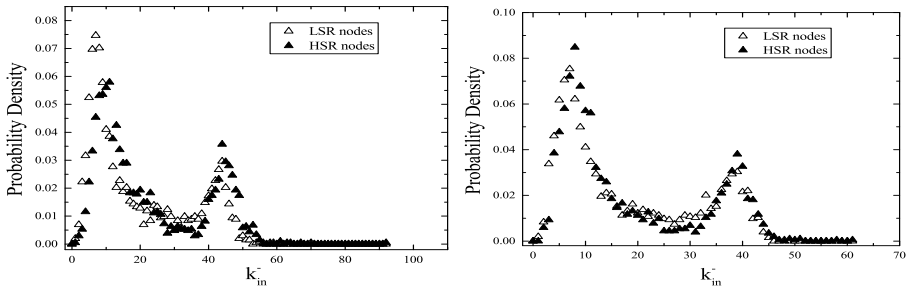
Figure 7.1: Division of spike rate into 10 decile groups in DIV11. For each box, red short line present the median Q_2 of groups, the lower quartile Q_1 and upper quartile Q_3 form the box. Define interquartile range ($IQR = Q_3 - Q_1$), the upper limit and lower limit (black line) are defined as the maximum and minimum within the non-anomalous range $[Q_1 - 1.5IQR, Q_1]$ and $[Q_3, Q_3 + 1.5IQR]$. Red cross means the data outside the range.



(a) DIV11 (b) DIV33

(c) DIV52 (d) DIV66

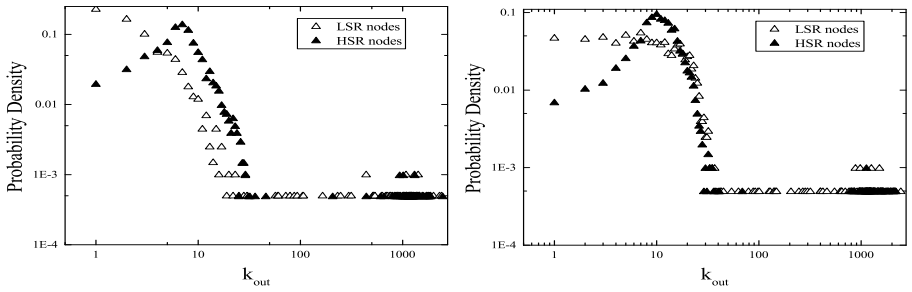
Figure 7.2: Distribution of k_{in}^{+} in HSR (solid) and LSR (open) for several DIVs.



(a) DIV11 (b) DIV33

(c) DIV52 (d) DIV66

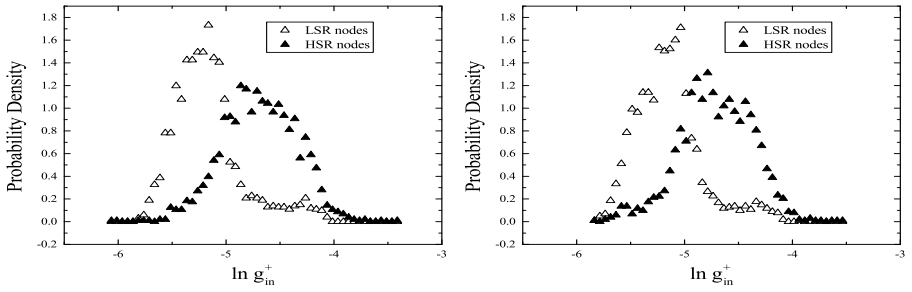
Figure 7.3: Distribution of k_{in}^{-} in HSR (solid) and LSR (open) for several DIVs.



(a) DIV11 (b) DIV33

(c) DIV52 (d) DIV66

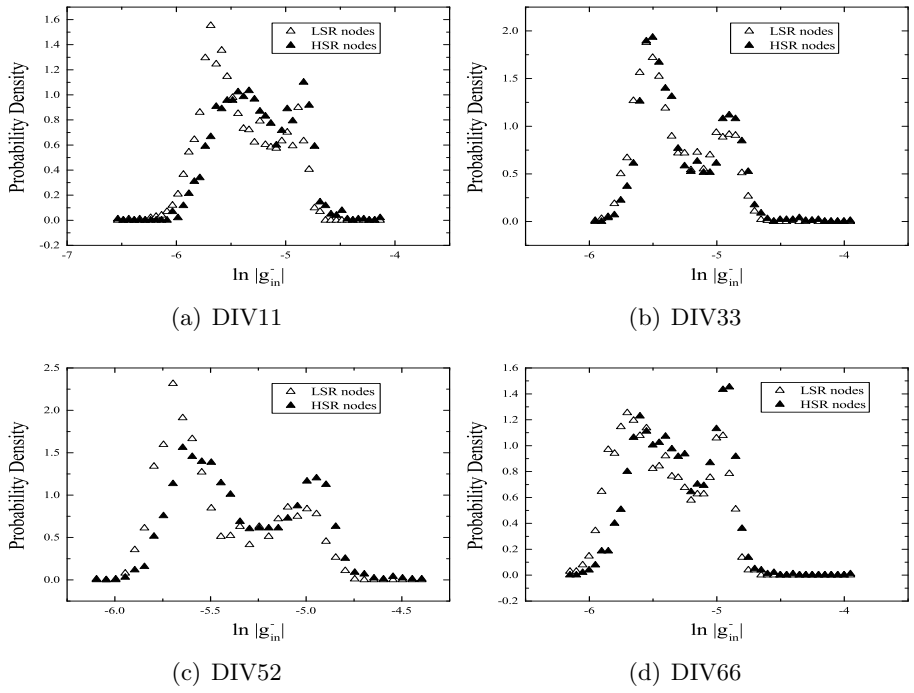
Figure 7.4: Distribution of k_{out} in HSR (solid) and LSR (open) for several DIVs.



(a) DIV11 (b) DIV33

(c) DIV52 (d) DIV66

Figure 7.5: Distribution of $\ln g_{in}^{+}$ in HSR (solid) and LSR (open) for several DIVs.



(a) DIV11 (b) DIV33

(c) DIV52 (d) DIV66

Figure 7.6: Distribution of $\ln|g_{in}^{-}|$ in HSR (solid) and LSR (open) for several DIVs.

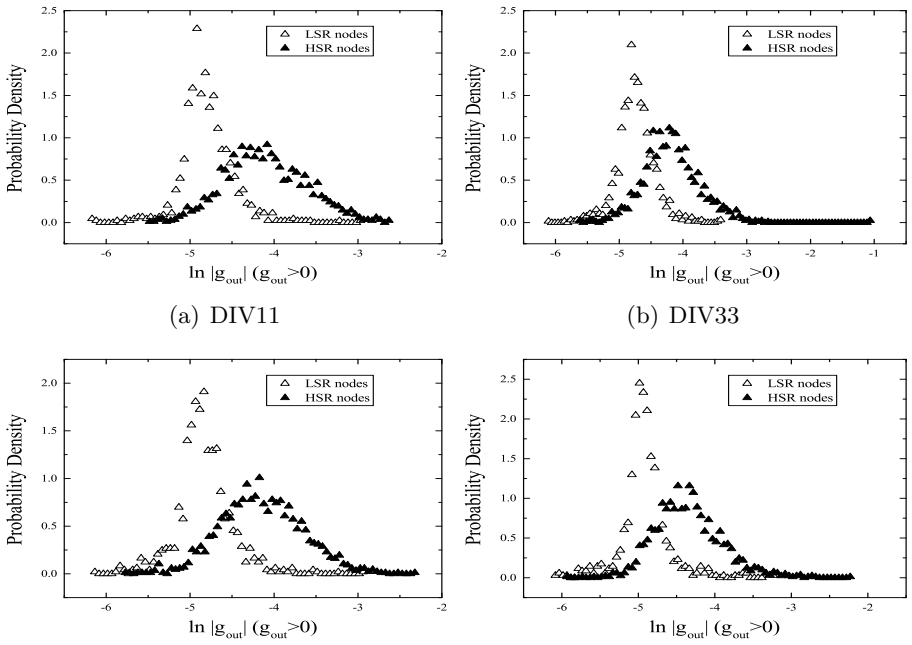
7.2.1 Excitatory incoming synaptic weight

To predict whether a node is HSR and LSR using a threshold value g_{th} for g_{in}^{+}. A node is predicted as HSR when having $g_{in}^{+} \geq g_{th}$, otherwise LSR. g_{th} is set as the multiples of $median(g_{in}^{+})$. Here Sensitivity and Specificity are used for measuring the precision

$$Sensitivity = \frac{\#\ HSR\ nodes\ with\ g_{in}^{+} \geq g_{th}}{\#\ of\ HSR\ nodes} \tag{7.1}$$

$$Specificity = \frac{\#\ LSR\ nodes\ with\ g_{in}^{+} < g_{th}}{\#\ of\ LSR\ nodes} \tag{7.2}$$

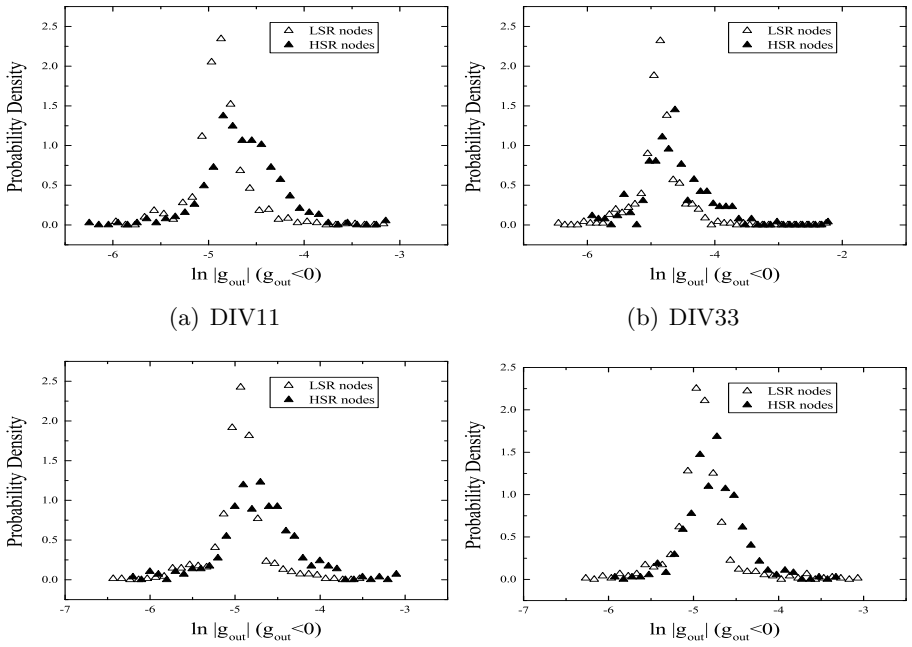
It is noted that $0 \leq Sensitivity, Specificity \leq 1$. Larger of Sensitivity and Specificity mean the better prediction results. By varying the threshold multiples, Figure 7.9 show the Receiver operating characteristic (ROC) curves for different DIVs. The ROC curves show Sensitivity versus 1-Specificity, which provides tools



(a) DIV11 (b) DIV33

(c) DIV52 (d) DIV66

Figure 7.7: Distribution of $\ln|g_{out}|(g_{out} > 0)$ in HSR (solid) and LSR (open) for several DIVs.



(a) DIV11 (b) DIV33

(c) DIV52 (d) DIV66

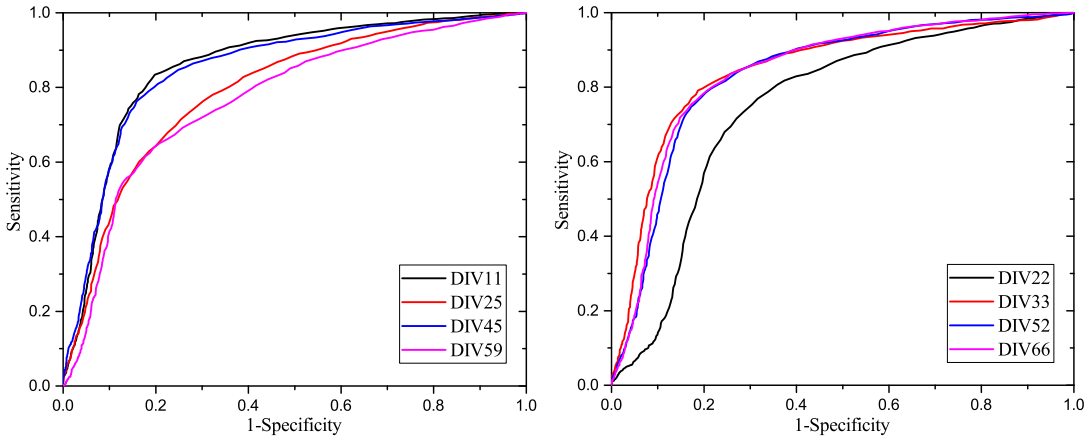
Figure 7.8: Distribution of $\ln|g_{out}|(g_{out} < 0)$ in HSR (solid) and LSR (open) for several DIVs.

to select possibly optimal models when choosing g_{in}^{+}. To weight the importance of Sensitivity and Specificity, we use two indexes to measure the results: Youden's J statistic [92] and Closest distance to perfect results (the distance from (0,1) in ROC curves)

$$Youden = Sensitivity + Specificity - 1 \tag{7.3}$$

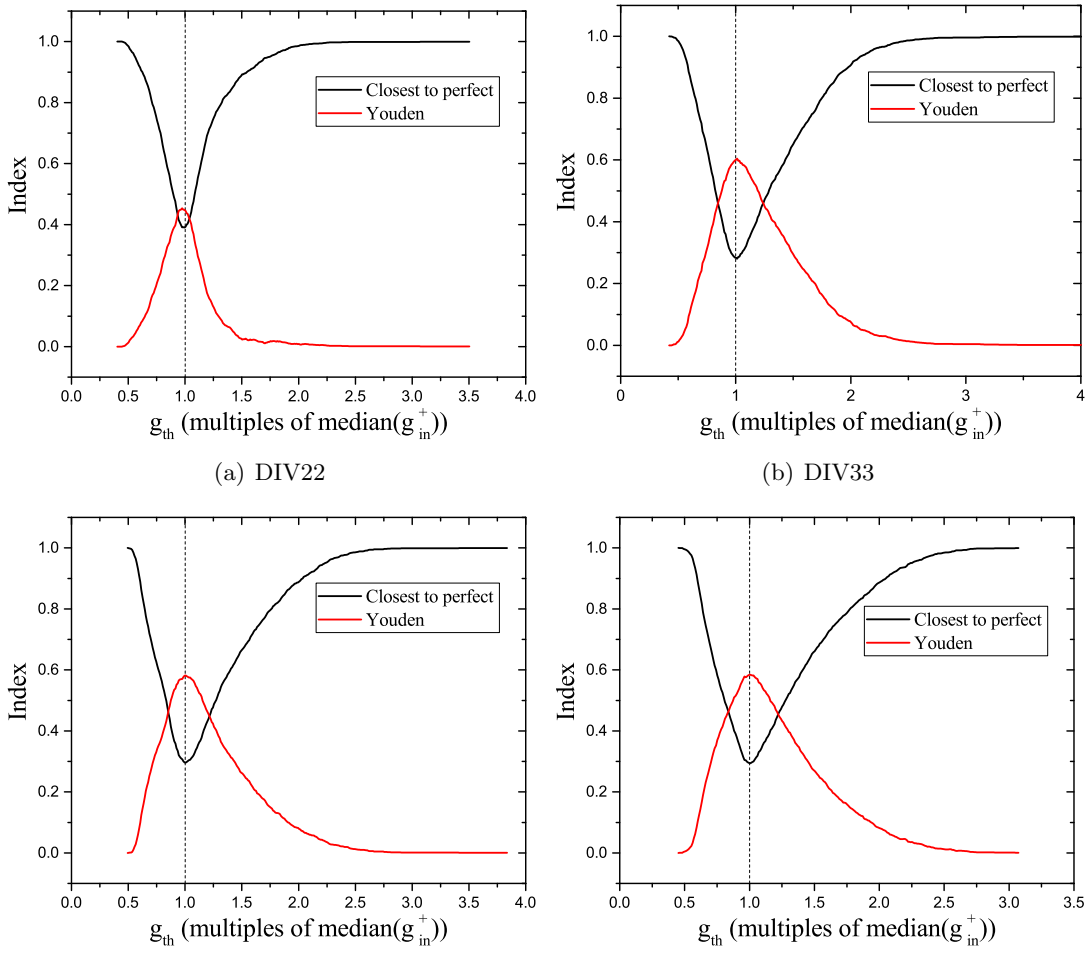
$$Closest\ distance = \sqrt{(1 - Sensitivity)^2 + (1 - Specificity)^2} \tag{7.4}$$

Both maximum Youden's J statistic and minimum Closest distance reflect the perfect prediction results. g_{th} is chosen for the best results. Figure 7.10 show the two indexes in different DIVs. Maximum Youden's J statistic and minimum Closest distance are obtained when the multiple is approximate to 1. So, we choose $g_{th} = median(g_{in}^{+})$ in predictions. Table 7.1 is the prediction results. Most of Sensitivity and Specificity are around 70%-80%, and Youden's J statistic is mostly around 0.4-0.6. The mean value of Youden's J statistic for 8 DIVs is 0.54. It is a good prediction result.



(a) DIV11,25,45,59 (b) DIV22,33,52,66

Figure 7.9: ROC curves for different DIVs. The color of curves are presented in legends and stand for different DIVs.



(a) DIV22

(b) DIV33

(c) DIV52

(d) DIV66

Figure 7.10: Youden's J statistic (red curve) and closest to perfect (black curve) in different g_{th}. The dashed line is x=1.

	DIV11	DIV22	DIV25	DIV33	DIV45	DIV52	DIV59	DIV66
Sensitivity	80.98%	72.19%	72.86%	79.81%	80.08%	78.83%	71.00%	79.06%
Specificity	81.71%	72.28%	72.93%	80.15%	80.36%	79.27%	71.42%	79.19%
Youden	0.627	0.445	0.458	0.600	0.604	0.581	0.424	0.582

Table 7.1: Sensitivity and Specificity as well as Youden's J statistic in predictions using g_{in}^{+} for 8 DIVs.

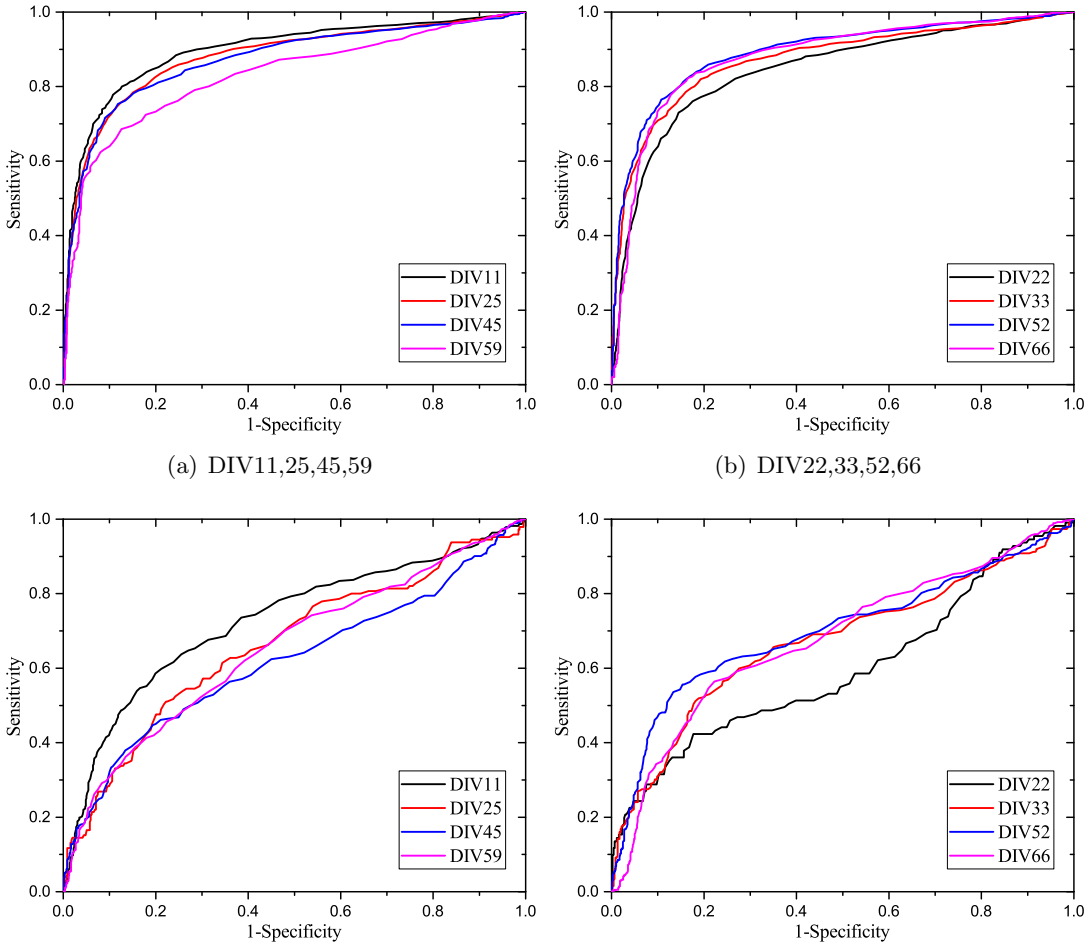
7.2.2 Outgoing synaptic weight

g_{out}, especially positive g_{out}, can be also used to predict whether the spike rate belongs to HSR or LSR. g_{out} are separated into positive and negative as well as 0, which stand for excitatory nodes, inhibitory nodes and no-outgoing-links nodes. Define a positive threshold g_{th}^{+} for positive g_{out} as multiples of $median(positive\ g_{out})$ and negative threshold g_{th}^{-} for negative g_{out} as multiples of $median(negative\ g_{out})$. Positive g_{out} that larger than or equal to g_{th}^{+} are regarded as HSR otherwise LSR. Negative g_{out} that smaller than or equal to g_{th}^{-} are regarded as HSR otherwise LSR. Nodes with no detectable outgoing links are predicted LSR because over 90% of them have a small spike rate. In summary, nodes with $g_{th}^{-} < g_{out} < g_{th}^{+}$ are predicted as LSR nodes, otherwise are predicted as HSR nodes. Nodes in higher absolute values have a higher chance to be HSR nodes.

Figure 7.11 is ROC curves for both positive and negative g_{out}. Obviously, positive g_{out} have better precision than negative g_{out}. Figure 7.12 show that the threshold are better set as $median(positive\ g_{out})$ and $median(negative\ g_{out})$ because Youden's J statistic and closest distance to the perfect result reach their best results in such thresholds in DIV33. Other DIVs have the same results. Prediction results are shown in Table 7.2. Similar as results of Table 7.1, Sensitivity and Specificity of predictions are mostly around 70%-80% and Youden's J statistic are around 0.4-0.6. The mean value of Youden's J statistic for 8 DIVs is 0.51, and the difference between Sensitivity and Specificity are obviously large. The prediction results are not as good as prediction using g_{in}^{+}.

7.2.3 k-NN algorithm

We noticed other features like k_{in}^{+} and k_{out} also have an obvious relation with spike rate. For better prediction, we study an algorithm, named k Nearest Neigh-



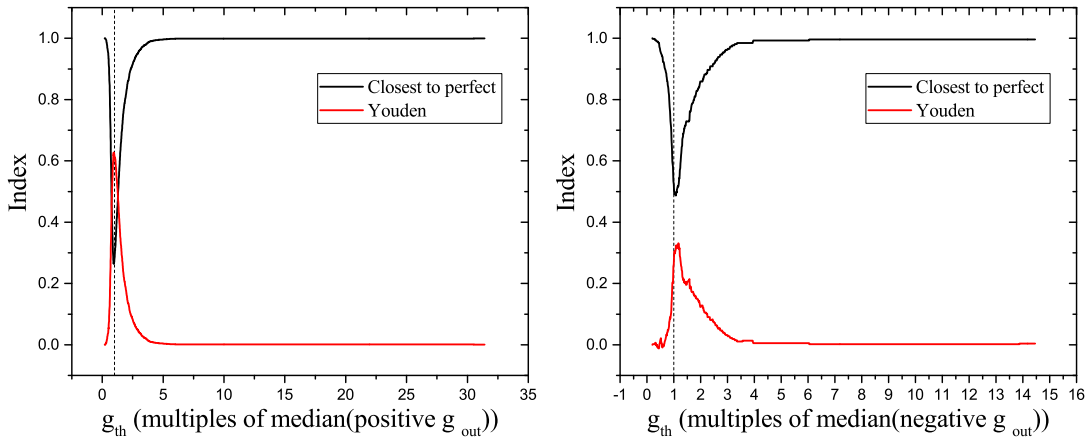
(a) DIV11,25,45,59 (b) DIV22,33,52,66

(c) DIV11,25,45,59 (d) DIV22,33,52,66

Figure 7.11: ROC curves for different DIVs. The upper figures are for positive g_{out}, the lower figures are for negative g_{out}. The color of curves are presented in legends and stand for different DIVs.

	DIV11	DIV22	DIV25	DIV33	DIV45	DIV52	DIV59	DIV66
Sensitivity	71.33%	72.19%	77.29%	74.22%	66.72%	71.51%	60.80%	66.15%
Specificity	83.04%	78.60%	79.10%	81.67%	83.65%	81.49%	81.34%	81.54%
Youden	0.544	0.508	0.564	0.559	0.504	0.530	0.421	0.477

Table 7.2: Sensitivity and Specificity as well as Youden's J statistic in predictions using g_{out} for 8 DIVs.



(a) positive g_{out} (b) negative g_{out}

Figure 7.12: Youden's J statistic (red curve) and closest to perfect (black curve) in different g_{th} of DIV33. The dashed line is x=1.

bors (k-NN) algorithm, to take full use of all network structure features in spike rate predictions. In pattern recognition, the k-NN algorithm is a non-parametric method used for classification and regression [93]. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression.

- In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbor.

- In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.

The k-NN algorithm is based on the assumption that similar classifications should have similar features. 10% of nodes, as well as their all 6 network, features information is needed for training and test the rest 90% of nodes' classifications.

Procedures of k-NN

The key idea of k-NN classification is that the close-distance in feature space has the same class membership. We use a weighted distance k-NN algorithm, which can consider the effect of distance. Procedures of k-NN are

- There are 6 independent features $k_{in}^{+}, k_{in}^{-}, k_{out}, g_{in}^{+}, g_{in}^{-}, g_{out}$. The number of chosen features $n = 6$ and $N = 4095$ is the number of nodes.

$$(x_1^{(1)}, x_1^{(2)}, \ldots, x_1^{(n)}), \ldots, (x_N^{(1)}, x_N^{(2)}, \ldots, x_N^{(n)}) \tag{7.5}$$

- Standardize each structure feature and scales each structure feature by mean and standard deviation, respectively

$$x_k^{(j)} \leftarrow \frac{x_k^{(j)} - \left\langle x^{(j)} \right\rangle}{std(x^{(j)})} \tag{7.6}$$

$$\left\langle x^{(j)} \right\rangle = \frac{1}{N} \sum_{i=1}^{N} x_i^{(j)}, \ std(x^{(j)}) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(x_i^{(j)} - \langle x^{(j)} \rangle \right)^2} \tag{7.7}$$

- In each DIV, randomly choose part of nodes' feature data (10%, 409 nodes) and their classification (HSR or LSR) as training data. The rest feature data are regarded as testing data. Our aim is to predict the classification of testing data in use of the knowledge of training data's feature value and classification as well as testing data's feature value. Set up a N-dimension feature space (e.g. $N = 6$) and place the nodes of training data into feature space.

- Put every node (set it $((x^{(1)}, x^{(2)}, \ldots, x^{(n)}))$) into the feature space with their classification "HSR" or "LSR". Calculate the Minkowski Distance

(L^p *norm*) between all other nodes

$$d(i) = ||x||_p = \left(\sum_{j=1}^{n} |x^{(j)} - x_i^{(j)}|^p \right)^{\frac{1}{p}} \tag{7.8}$$

- Select the minimum-k d (nearest nodes) and focus on them. Consider the distance weight $w(i) = 1/d(i)$. Calculate the summation of weight for each classification

$$W_{HSR} = \sum_{i=1, i \in HSR}^{k} w(i), \; W_{LSR} = \sum_{i=1, i \in LSR}^{k} w(i) \tag{7.9}$$

  If $W_{HSR} \geq W_{LSR}$, the newly added node is regarded as HSR nodes, otherwise is regarded as LSR nodes.

- After all test data are added, calculate the Sensitivity and Specificity of classification method of test data as follows

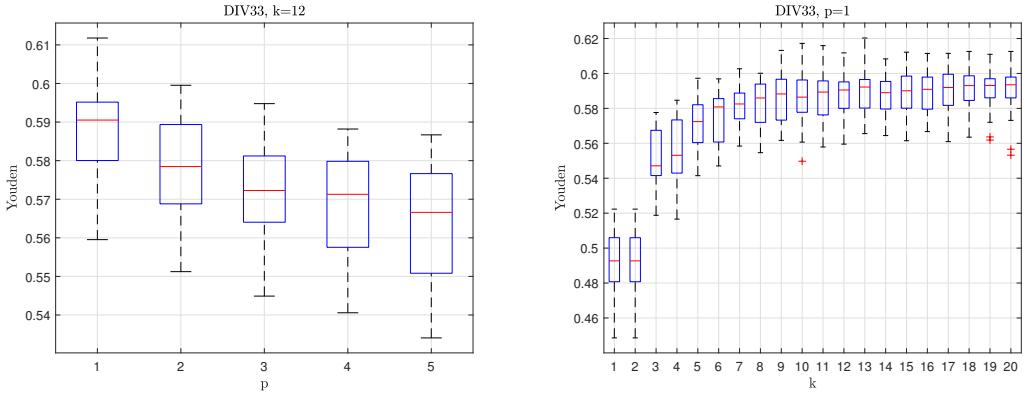
$$Sensitivity = \frac{\#\ truly\ predicted\ HSR\ nodes}{\#\ HSR\ nodes} \tag{7.10}$$

$$Specificity = \frac{\#\ truly\ predicted\ LSR\ nodes}{\#\ LSR\ nodes} \tag{7.11}$$

Influence of k and p

It is known that hyperparameters would affect the predictions. We should vary the hyperparameters k and p as positive integers for the best results by maximizing the Youden's J statistic. p in L_p *norm* are varied from 1 to 5 and k from 1 to 20. I notice that the k used for weighting the new adding node. So, in general, the larger k causes better results. However, for the good time consuming when calculation and the weight for nodes that are far away from the new adding node are very small. We only need to fix a proper k such as $k = 12$. In each case,

30 sets of 10% data are used for training, obtaining 30 Youden's J statistic for boxplots of DIV33 in Figure 7.13. In Figure 7.13 (a), obviously $p = 1$ reaches more accurate results for fixed k. And in Figure 7.13 (b), small k have bad results because too few nearby nodes may lead to wrongly recognition for fixed p. But too large k will have little effect on the precision. In this way, the mean values of all DIVs are shown in Figure 7.14, indicating that $k = 12, p = 1$ is a good selection for prediction.



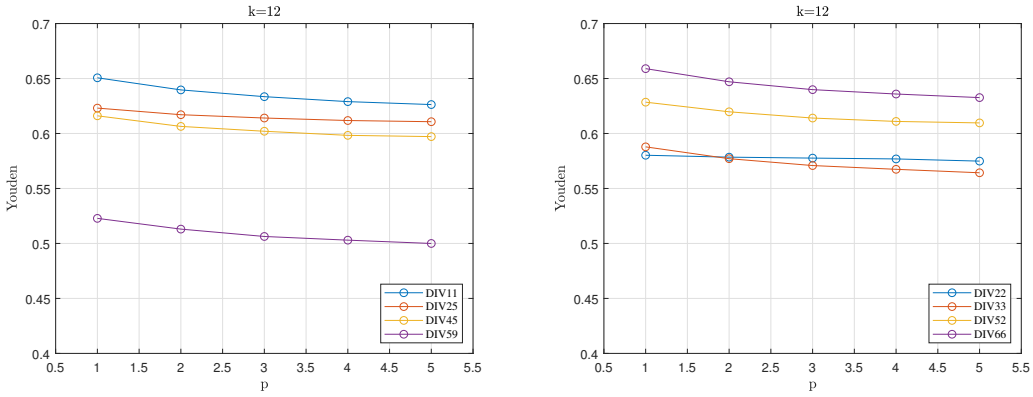
(a) Youden versus p in $k = 12$

(b) Youden versus k in $p = 1$

Figure 7.13: Boxplots of Youden's J statistic in fixed $k = 12$ or fixed $p = 1$ of DIV33. For each box, red short line present the median Q_2 of groups, the lower quartile Q_1 and upper quartile Q_3 form the box. Define interquartile range ($IQR = Q_3 - Q_1$), the upper limit and lower limit (black line) are defined as the maximum and minimum within the non-anomalous range $[Q_1 - 1.5IQR, Q_1]$ and $[Q_3, Q_3 + 1.5IQR]$. Red cross means the data outside the range.

Classification results

Under the hyperparameters $k = 12$ and $p = 1$, we calculate the mean value of Sensitivity, Specificity and Youden's J statistic results for 30 different training datasets in each DIV (Table 7.3). Sensitivity and Specificity of predictions are mostly around 75%-85% and Youden's J statistic is around 0.55-0.65. Mean value



(a) DIV11,25,45,59

(b) DIV22,33,52,66

(c) DIV11,25,45,59

(d) DIV22,33,52,66

Figure 7.14: Mean value of Youden's J statistic in fixed $k = 12$ or fixed $p = 1$ for different DIVs. The curves indicate the trend of mean value of Youden's J statistic within same parameters. The color of curves are presented in legends and stand for different DIVs.

of Youden's J statistic for 8 DIVs is 0.61, which are larger than results of prediction using g_{in}^{+} and g_{out}, showing the highest accuracy in predictions.

	DIV11	DIV22	DIV25	DIV33	DIV45	DIV52	DIV59	DIV66
Sensitivity	76.25%	75.75%	76.30%	75.29%	74.85%	75.13%	70.34%	78.12%
Specificity	88.81%	82.27%	86.01%	83.49%	86.75%	87.72%	81.95%	87.77%
Youden	0.651	0.580	0.623	0.588	0.616	0.628	0.523	0.659

Table 7.3: Sensitivity and Specificity as well as Youden's J statistic in predictions for the mean of 30 different training datasets using k-NN algorithm for 8 DIVs.

Regression with k-NN

As discussed above, the k-NN algorithm can also do the regression work. That is, the prediction of spike rate, not only classified into HSR or LSR. The procedures are like classification procedures except for the procedure of spike rate calculations:

- Put every node (set it $((x^{(1)}, x^{(2)}, \ldots, x^{(n)}))$) into the feature space with their spike rate number $spkr$. Calculate the Minkowski Distance ($L^p norm$) between all other nodes

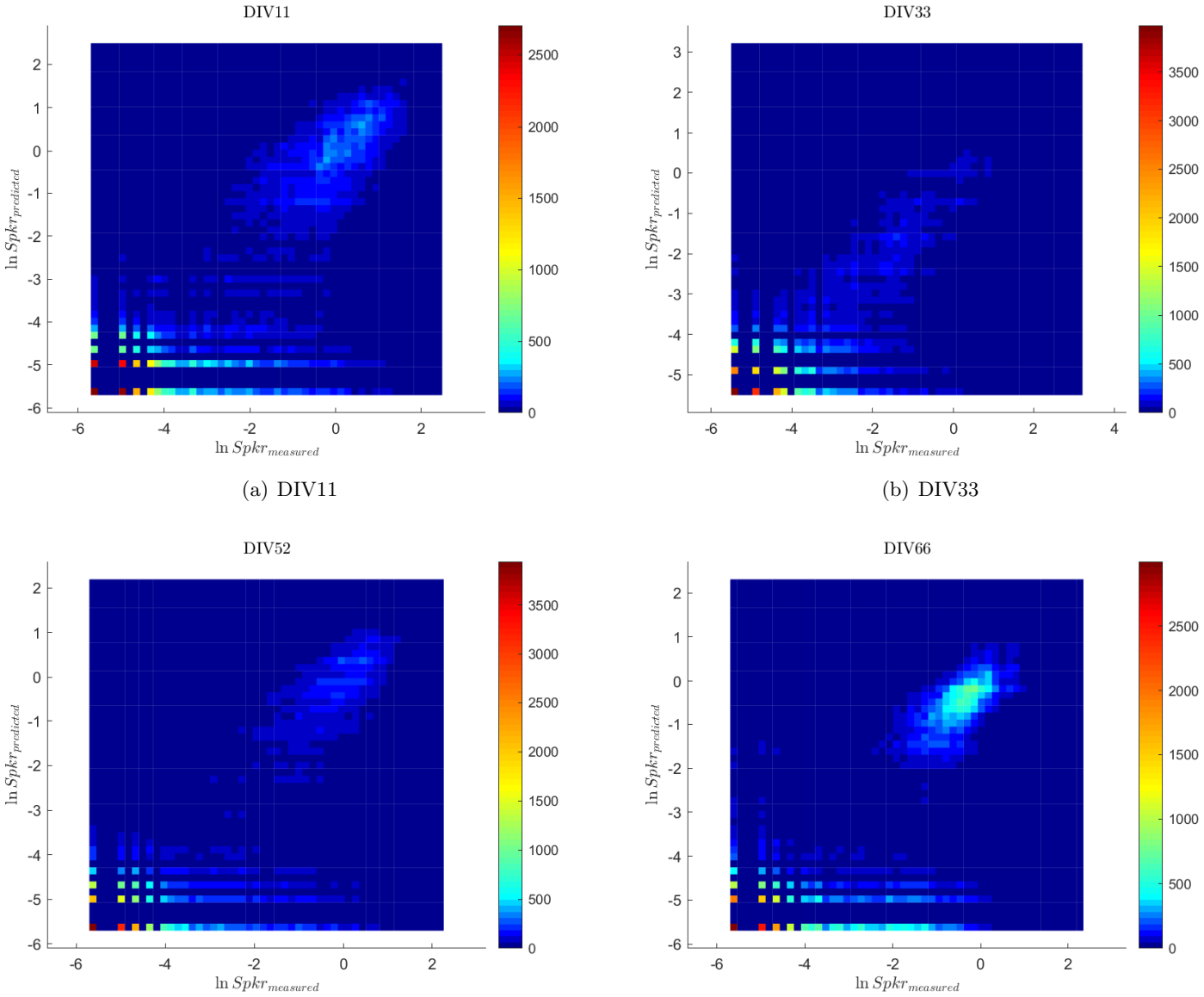
$$d(i) = ||x||_p = \left(\sum_{j=1}^{n} |x^{(j)} - x_i^{(j)}|^p \right)^{\frac{1}{p}} \tag{7.12}$$

- Select the minimum-k d (nearest nodes) and focus on them. Consider the distance weight $w(i) = 1/d(i)$. Make a summation of them $W = \sum_{i=1}^{k} w(i)$. So the new added node's spike rate is given by

$$Spkr_{predicted}^{new\ adding} = \sum_{i=1}^{k} \frac{w(i)}{W} Spkr_{measured}(i) \tag{7.13}$$

Since most of $Spkr_{measured}$ and $Spkr_{predicted}$ are very small and it is hard to

illustrate by plotting $Spkr_{measured}$ versus $Spkr_{predicted}$, Figure 7.15 show joint histograms between $\ln Spkr_{predicted}$ and $\ln Spkr_{measured}$ throughout 30 different training datasets for different DIVs. Small $\ln Spkr_{measured}$ have small $\ln Spkr_{predicted}$ and large $\ln Spkr_{measured}$ would be also predicted large. The figures all generally follow the $y = x$ in all DIVs.



(a) DIV11

(b) DIV33

(c) DIV52

(d) DIV66

Figure 7.15: Joint histograms between ln $Spkr_{predicted}$ and ln $Spkr_{measured}$ for different DIVs.