

目录

1	Introduction	2
2	方法	3
2.1	核密度估计	3
2.2	JS Loss	4
2.3	JSD Loss functions Definition	4
2.4	2D Kernel density estimation	4
3	风功率曲线	5
3.1	风速的分布	6
4	实验设置	6
4.1	数据集描述	6
4.2	风功率曲线的计算方法	6
4.3	在训练数据中添加高斯噪音	7
4.4	使用气象预报数据	7
5	实验	8
5.1	数据量的影响	8
5.2	带宽选择的影响	8
5.3	训练方式	9
5.4	$N(x, x)$ 噪音	10
5.5	气象预报	11
6	Appendix	11

风机发电量预测实验报告

A PREPRINT

2022 年 9 月 1 日

1 Introduction

作为一种从风力中获取的能量，风能具有不连续，多变，不可调度的特点。同时容易受到短期极端天气如暴风的影响。同时如果没有储能系统，风电在被生产出来以后必须被马上使用。基于以上特点，风电的经济价值很大程度上取决于风功率预测地准确性。风功率预测也可以提供未来一段时间内的发电量信息。风力发电量预测是风能资源的整合和运营中最为重要的环节。风能的预测主要分为长期，中期和短期。中短期预测的需求体现在风场的市场运营和配电方面、基于不同的时间窗口，风力发电量中有很多因素需要考虑，1.短期气流影响。2.月度、年度的气候。3.天气尺度，通常为6小时到1周。通常将间隔小于30分钟的预测称为超短期预测。

为了优化风场和电网之间的电力调度，有多种方法被提出来。大体可以分为三类，一种是基于物理模型的方法，包括各种数值天气预报，通过对区域气候气象进行建模，考虑各种初始状态和边界信息，计算出未来的风速风向，间接预测风机功率。二种是基于统计的方法，构建出各种统计模型，得到未来发电功率和已知特征之间的关系。三是基于数据的方法，包括各种数据驱动模型如ANN等，利用机器学习历史发电信息用挖掘出有效的模式，结合天气预报，能提供尽量准确发电量预测。第四种是混合方法，结合物理模型和机器学习的模型，互相补充。

在基于机器学习的风机发电量预测当中，通常的输入特征有各种气象数据，历史发电数据，模型会在训练过程中，利用训练数据建立输入与输出的关系。但是在预测环节，模型可能产生一些不符合物理规律的结果，如小于零的物理量，加入物理限制则可以消除这些误差。风功率曲线，作为风机的固有物理性质如果能参与到神经网络的训练当中，则可以很好地消除训练过程当中不符合物理规律的输出。常规的方法里，风功率曲线通过统计方法得到，结果是在计算过程当中无法产生梯度，而机器学习需要通过梯度回传的原理优化神经网络，这样造成风功率曲线的特征无法参与到机器学习的训练当中。

本研究通过基于核密度估计的方法计算风功率曲线，利用JS散度计算模型输入的风功率曲线和真实风功率曲线之间的误差，首次将风功率曲线引入到深度学习当中。在训练过程中，不符合风功率曲线的结果逐渐减少，相较于常规的MSE Loss训练的网络，新的网络中在输出预测功率时严格符合预先计算的风功率曲线，提高了结果的可靠性。

2 方法

2.1 核密度估计

核密度估计 (Kernel Density Estimation), 又称为Parzen's window[Parzen, 1962]. 在概率论中用来估计未知的密度函数, 属于非参数检验方法之一。核密度估计函数采用平滑的峰值函数来拟合观察到的数据点, 从而对真实的概率分布曲线进行模拟。令 (x_1, x_2, \dots, x_n) 是独立且同分布的样本, 这些样本是从任何给定点 x 处具有未知密度 f 的某个单变量分布中抽取的。为了估计这个函数 f 的形状, 它在 x 点处的核密度估计是:

$$\hat{f}_x = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (1)$$

其中 K 被称为核函数, h 被称为带宽, n 是样本量, 一般采用高斯核函数, 其形式为:

$$K\left(\frac{x - x_i}{h}\right) = \left(\frac{1}{\sqrt{2\pi}}\right) e^{-\frac{(x - x_i)^2}{2h^2}} \quad (2)$$

核密度估计的效果和核函数的选择关系不大, 主要受带宽选择的影响[Sheather, 2004, Scott, 2015]。带宽的选择对估计结果有很大影响, 带宽也被称为平滑系数, 如果带宽选择小, 密度估计函数中会有很多起伏, 如果带宽选择太大, 则会缺失大部分的细节, 只有选择了合适的带宽, 才能使密度估计反映真实的分布。有很多种确定带宽的方法, 常用的有rule of thumb, least square cross validation[Rudemo, 1982, Bowman, 1984, Stone, 1984], biased cross-validation[Scott and Terrell, 1987], 和plug-in method[Woodroffe, 1970, Sheather and Jones, 1991]。总的来说, 不同带宽估计方法的最终目的是减小与目标分布的预测误差。一般用mean integrated squared error (MISE)来衡量概率估计的准确程度。

如果将MISE作为优化目标, 最终目的是找到一个 h

$$h_{MISE} := \underset{h>0}{\operatorname{argmin}} MISE[\hat{f}(\cdot; h)] \quad (3)$$

在本次研究中, h 的计算方法如下, 其中 λ 是一个可调参数, 通过实验确定。

$$h = \lambda \hat{\sigma} n^{-\frac{1}{5}} \quad (4)$$

$$\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N - 1}} \quad (5)$$

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad (6)$$

2.2 JS Loss

由于作为优化目标的风功率曲线是一个分布，为了计算模型分布和真实分布之间的误差，引入了JS散度。Jensen-Shannon divergence(JSD)可以衡量两个概率密度之间的差异，相对于对于概率分布 $P(x)$ 和 $Q(x)$ ：

$$JS(P\|Q) = \frac{1}{2}KL(P(x)\|\frac{P(x)+Q(x)}{2}) + \frac{1}{2}KL(Q(x)\|\frac{P(x)+Q(x)}{2}) \quad (7)$$

JSD是Kullback-Leibler divergence(KLD)[Kullback and Leibler, 1951]的一个扩展，公式中 KL 代表Kullback-Leibler divergence，可以用来度量两个密度分布之间的差异程度。KL距离也被称为信息距离或则相对熵，等价于像个概率分布的信息熵的差值。假设 P 和 Q 为两个概率分布。

$$KL(P\|Q) = \int_{\mathbb{R}^d} p(x) \log \frac{p(x)}{q(x)} dx \quad (8)$$

与KLD不同，JSD是一个有界的度量，值域范围是 $[0, 1]$ 。如果两个分布完全相同，则JSD等于0。同时JSD具有对称性。但是JS散度有一个缺点，如果两个分布完全没有重叠，JSD会成为一个固定值（此部分等待细化），在计算过程中无法产生梯度，造成梯度消失。而在神经网络的训练过程中，网络在初始状态下输出的分布与真实分布可能完全没有重叠，一个解决办法是动态地调节Loss中JSD Loss的比例，在训练初期依靠MSE Loss进行梯度更新，在后期再慢慢提高JSD Loss的比例。

在确定 \mathcal{L}_{JS} 的比例 α 时，有两种方法，一种是基于模型输出结果的标准差，在每个迭代步更新 α ，因为在训练初期，模型输出结果的标准差接近0，到训练接近完成之后，模型输出的标准差接近1，这样可以构造一个从小到大的变量来控制JSD Loss的比例。一种是使用固定值 c ，这样在训练过程中需要将 c 设置位一个较小的数，防止训练初期无法得到足够的梯度进行模型更新，提升训练速度。在实验 $N(x, x)$ 噪音和 $N(0, x)$ 噪音中使用的是固定值，在实验预报数据中使用的是模型输出结果的标准差。

$$\mathcal{L}_{total} = \alpha \mathcal{L}_{JS} + \mathcal{L}_{MSE} \quad (9)$$

$$\alpha = \sigma_{modelout} \quad (10)$$

2.3 JSD Loss functions Definition

JS Loss的计算需要用到两个分布，图 2.3中左边真实风速与模型预测功率的风功率曲线，右边是通过历史数据计算出来的风功率曲线。计算风功率曲线的风速和功率是训练过程中所用到的所有数据。研究中使用核密度估计的方法计算风功率曲线。因为训练过程中，模型输出的功率不断在变化，导致模型输出功率的方差在不断变化，为了准确计算出模型输出的核密度估计，需要在每个迭代步更新带宽 h 。

2.4 2D Kernel density estimation

对于风功率曲线这种二维分布，在计算最终的的概率密度 \hat{f} 时：

$$\hat{f}_h(x, y) = K_h(x - x_i)K_h(y - y_i) \quad (11)$$

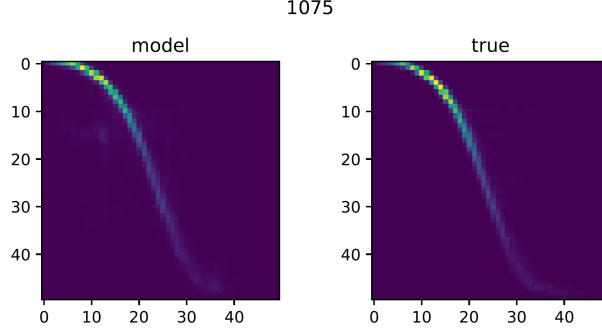


图 1: 模型风功率曲线和真实风功率曲线

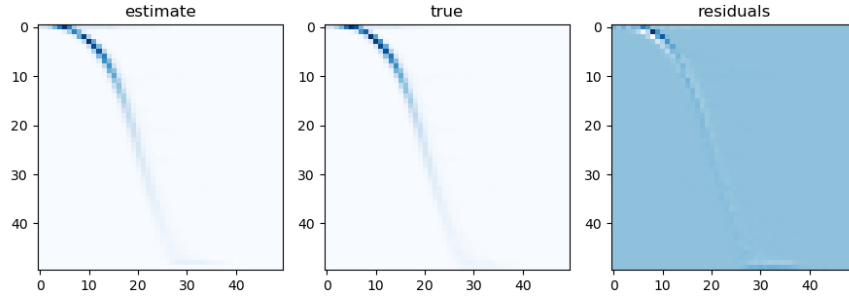


图 2: 核密估计的风功率曲线和基于统计的风功率曲线

3 风功率曲线

风机将风能转化为电能，风功率曲线则衡量了风机将风能转化为电能的能力。风功率曲线指风机输出功率与风速的对应曲线，其中不仅包含了风机产生能量的信息，同时也反映了风机的机械电子方面的能力。一个风机的发电量很大程度上受风速的影响，不同型号的风机有着不同的性能，通过风功率曲线，可以直观地了解风机的功率产出而不必知道风机的技术细节。风机供应商会提供基于工业标准 IEC 61400-12-1 测量的风机功率曲线。风功率曲线可以以一种函数的方式来表示某一高度风速和风机发电量的关系。驱动风机发电的最小风速被称为 cut-in speed，当风机发电量达到最大后，为了保护风机，即使风速继续增大，风机产生的电量不再增加。风机能正常工作的最大速度被称为 cut-off speed。理论上风机的发电量可以由如下公式确定

$$P = \frac{1}{2} \rho A (v \cdot \sin \varphi)^3 C_p = \frac{B}{2T \cdot R} A (v \cdot \sin \varphi)^3 C_p \quad (12)$$

其中 ρ 为空气密度， A 为叶片扫掠面积， v 为风速， φ 为叶轮与风向之间的夹角， C_p 是功率系数， B 为气压， T 为空气开氏绝对温度， R 为气体常数。[Lydia et al., 2014] 在论文中总结了风功率曲线的参数和非参数建模方法，参数模型包括 Linearized Segmented Model, Polynomial Power Curve, Maximum Principle Method, Dynamical Power Curve, Probabilistic Model, Ideal power curve, 4-parameter Logistic Function, 5-parameter Logistic Function。非参数模型包括 Copula power curve Cubic Spline Interpolation, Neural Networks, Fuzzy methods, Datamining algorithms。

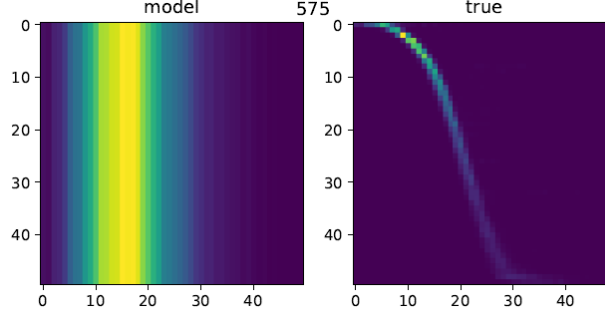


图 3: 初始状态的模型风功率曲线对比真实风功率曲线

3.1 风速的分布

在风速预测中，确定风速的分布有着重要的作用。某个区域的风速跟当地的气候紧密相关，并且有很强的周期性，在一段时间中也具有统计上的稳定性。通过分析长期的风速观测数据，很多种参数化的模型被提出来描述风速的分布，比如Weibull distribution[Garcia et al., 1998], lognormal distribution[Jangamshetti and Ran, 2001]。这些模型在一些地方的的风速概率密度估计中取得了很好的效果。核密度估计作为一种非参数化的概率密度估计方法，不需要预先知道数据的分布情况，也不需要数据的分布形态做出任何估计，有很好的灵活性和稳定性，同时在模型的精度上也有保证。近期的一些研究如[Qin et al., 2011, Hu et al., 2017, Jeon and Taylor, 2012]使用了核密度估计对风速的分布建模，在文章中讨论了核密度估计中不同的带宽估计方法对风速分布估计效果的影响。

4 实验设置

4.1 数据集描述

整个数据集由25台风机在2020年整年的数据组成，包括桨距角，发电机转速，平均风速，有功功率。使用1月到10月的数据进行训练，11月和12月的数据进行测试。训练中输入数据长度为4x24h,包括风速、桨距角度和风机有功功率，其中风力数据为三天的历史风力数据加上未来24小时的数据,未来24小时数据可以为添加了噪音的真实风速或气象预报数据。测试时输入数据时11月和12月的数据，其中风力数据为整段的噪音数据或真实气象预报数据。数据集的情况如下

4.2 风功率曲线的计算方法

由于风功率曲线是风机固有的物理性质，与风机的型号还有使用年限有关。同一风场中的风机往往有不同的风功率曲线，为了简化计算流程，在对比了平均风功率曲线和单个风功率曲线之间的查以后，本文采用平均风功率曲线作为目标风功率曲线，即将所有的风速和功率信息来计算风功率曲线，而不是利用每个风机的历史发电量和历史风速来计算不同的风功率曲线。这样的好处是在计算JSD Loss时只需要计算模型输出的风功率曲线和平均分功率曲线，而不是每一台风机都计算一次，大大减少了计算量，同时保证了计算的准确性。

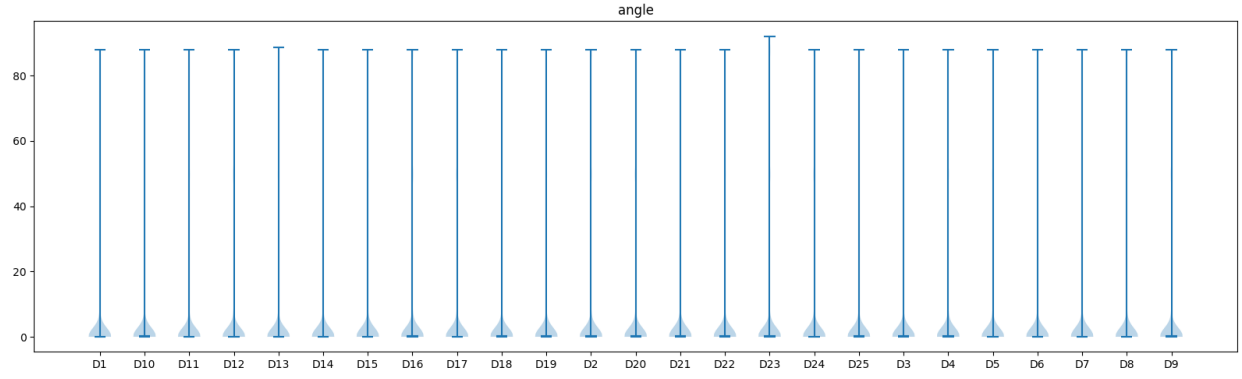


图 4: 桨距角的分布

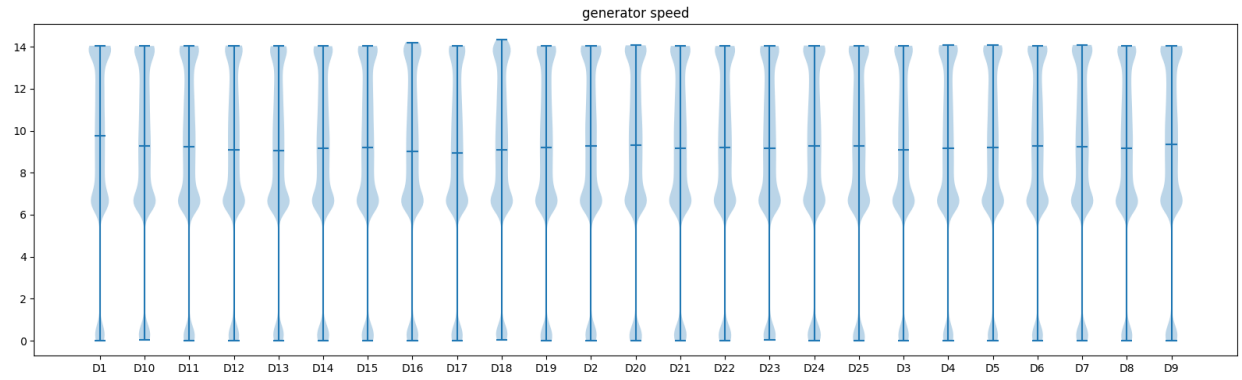


图 5: 发电机转速的分布

4.3 在训练数据中添加高斯噪音

添加了两种噪音，一种是 $N(0, x)$ ，一种是 $N(x, x)$ ，图 8 是添加了 $N(x, x)$ 噪音后的数据情况

4.4 使用气象预报数据

在使用气象预报数据时，训练中的风速数据是 $3 \times 24\text{h}$ 的历史风速数据加上24的预报数据，时间精度为15min，与真实功率的风功率曲线见图 9

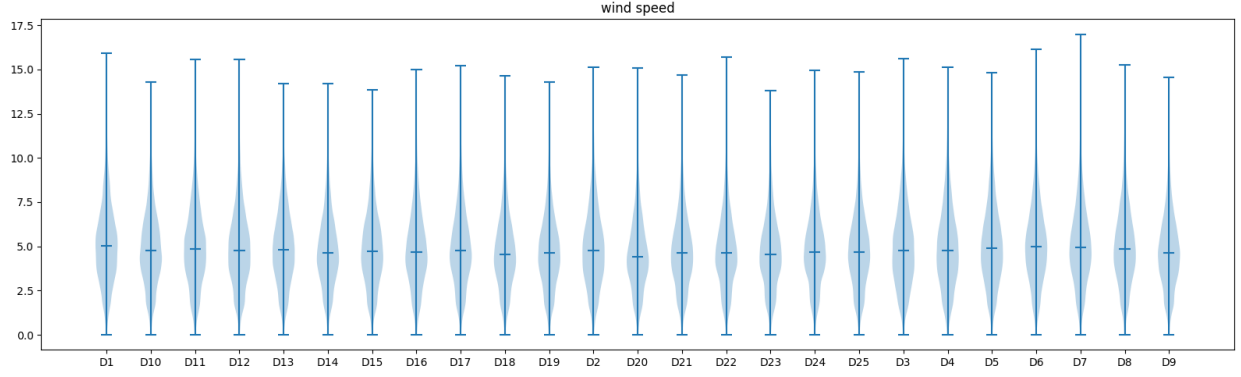


图 6: 平均风速的分布

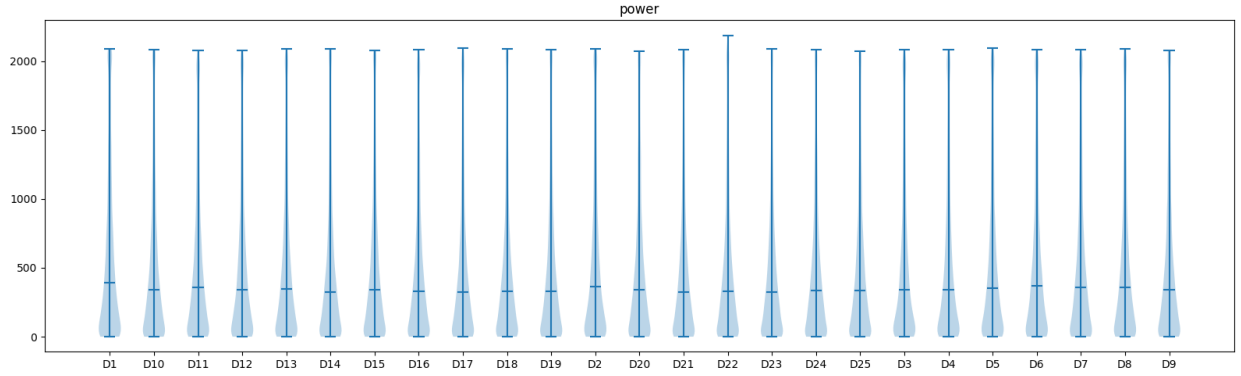


图 7: 有功功率的分布

5 实验

5.1 数据量的影响

训练数据的多少不仅会影响模型的表现，也会影响风功率曲线的计算效果，实验中使用了4, 6, 8, 10个月的历史数据作为训练数据，测试数据均为两个月。对比了不同数据量下模型的性能。在神经网络的训练当中，最终的模型效果受训练数据量的影响非常大。同时，在风功率计算过程中，由于不同风机的风功率曲线会有差异，不同时间段内的风功率曲线也会有变化。这些不同的数据，在核密度估计的计算过程中会产生不同的影响。

5.2 带宽选择的影响

早先的方案采取动态带宽，因此不同的带库可能会对结果产生影响。在最终的方案中，使用了固定带宽的方法，一开始模型的预测可能集中在一个很小的数值范围，但经过训练之后，模型的输出会多样化。

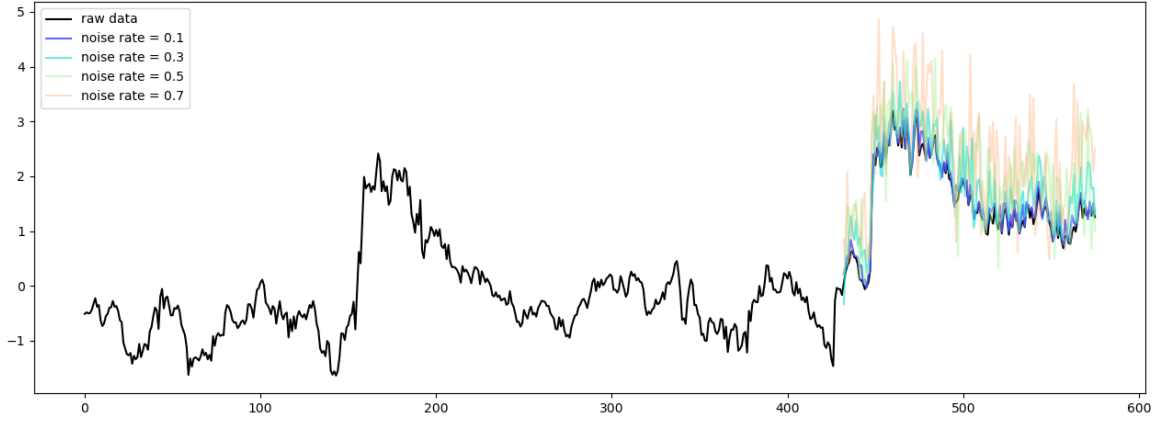
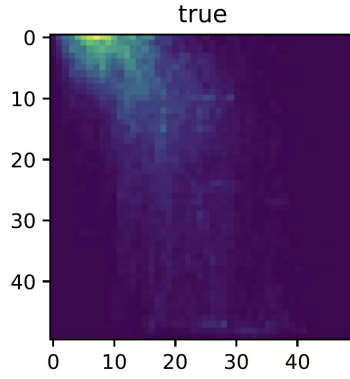
图 8: 带有 $N(x, x)$ 噪音的风速

图 9: 预报数据的风功率曲线

5.3 训练方式

由于最终的loss是由MSE和JS相加，在神经网络的迭代过程中，会对JSD Loss的比例进行动态调节。依次将 x 设置为0.1, 0.3, 0.5, 0.7

	Epoch	0				0.1				0.3				0.5				0.7			
		LSTM	JS			LSTM	JS			LSTM	JS			LSTM	JS			LSTM	JS		
N(0, 0.1)	10	0.0368	0.0322	14.37		0.0430	0.0386	11.42		0.0957	0.0868	10.25	0.1920	0.1734		10.72	0.3193	0.2858		11.72	
	20	0.0372	0.0307	21.18		0.0441	0.0374	17.94		0.0989	0.0879	12.54	0.1991	0.1764		12.85	0.3315	0.2944		12.59	
	30	0.0329	0.0300	9.57		0.0400	0.0371	7.85		0.0950	0.0883	7.57	0.2031	0.1857		9.35	0.3445	0.3113		10.65	
	40	0.0315	0.0301	4.58		0.0392	0.0373	5.16		0.0991	0.0911	8.77	0.2129	0.1910		11.45	0.3630	0.3214		12.95	
	50	0.0327	0.0319	2.63		0.0405	0.0392	3.30		0.1014	0.0948	6.90	0.2194	0.1996		9.92	0.3752	0.3351		11.97	
N(0, 0.3)	Epoch	0				0.1				0.3				0.5				0.7			
	10	0.0442	0.0374	18.19		0.0470	0.0413	13.99		0.0748	0.0744	0.60	0.1282	0.1375	-6.74	0.2050	0.2217	-7.52			
	20	0.0454	0.0356	27.34		0.0489	0.0401	22.11		0.0768	0.0733	4.87	0.1314	0.1341	-2.02	0.2110	0.2170	-2.76			
	30	0.0417	0.0372	12.13		0.0451	0.0416	8.49		0.0720	0.0743	-3.05	0.1300	0.1375	-5.47	0.2169	0.2231	-2.77			
	40	0.0412	0.0376	9.45		0.0443	0.0416	6.47		0.0721	0.0746	-3.39	0.1317	0.1360	-3.15	0.2254	0.2193	2.77			
	50	0.0423	0.0385	9.76		0.0454	0.0423	7.35		0.0726	0.0738	-1.61	0.1327	0.1340	-0.96	0.2292	0.2167	5.79			
N(0, 0.5)	Epoch	0				0.1				0.3				0.5				0.7			
	10	0.0604	0.0466	29.73		0.0627	0.0493	27.00		0.0791	0.0732	7.95	0.1131	0.1199	-5.68	0.1613	0.1820	-11.35			
	20	0.0589	0.0464	26.81		0.0612	0.0498	22.95		0.0767	0.0734	4.52	0.1082	0.1163	-6.98	0.1550	0.1727	-10.25			
	30	0.0589	0.0463	27.04		0.0608	0.0498	22.11		0.0750	0.0741	1.27	0.1053	0.1165	-9.58	0.1537	0.1703	-9.70			
	40	0.0616	0.0443	38.93		0.0629	0.0479	31.48		0.0762	0.0744	2.45	0.1042	0.1136	-8.26	0.1522	0.1603	-5.05			
	50	0.0626	0.0439	42.52		0.0646	0.0478	35.05		0.0766	0.0723	5.97	0.1057	0.1097	-3.62	0.1533	0.1541	-0.55			
N(0, 0.7)	Epoch	0				0.1				0.3				0.5				0.7			
	10	0.0768	0.0620	23.84		0.0783	0.0639	22.57		0.0896	0.0818	9.53	0.1127	0.1147	-1.73	0.1457	0.1596	-8.69			
	20	0.0770	0.0562	37.12		0.0781	0.0591	32.11		0.0882	0.0795	10.86	0.1083	0.1146	-5.46	0.1371	0.1553	-11.70			
	30	0.0820	0.0504	62.79		0.0831	0.0542	53.19		0.0908	0.0788	15.28	0.1084	0.1129	-4.01	0.1366	0.1515	-9.85			
	40	0.0881	0.0470	87.38		0.0889	0.0513	73.26		0.0963	0.0772	24.62	0.1120	0.1095	2.36	0.1374	0.1454	-5.53			
	50	0.0903	0.0475	89.93		0.0913	0.0525	74.09		0.0974	0.0784	24.18	0.1121	0.1091	2.77	0.1357	0.1438	-5.69			

图 10: 使用 $N(0, x)$ 噪音

5.4 $N(x, x)$ 噪音

依次将x设置为0.1, 0.3, 0.5, 0.7

	Epoch	0			0.1			0.3			0.5			0.7		
		LSTM	JS		LSTM	JS		LSTM	JS		LSTM	JS		LSTM	JS	
N(0.1, 0.1)	10	0.0310	0.0318	-2.54	0.0439	0.0437	0.46	0.1929	0.1794	6.99	0.4929	0.4519	8.33	0.9051	0.8286	8.46
	20	0.0310	0.0300	3.33	0.0439	0.0446	-1.52	0.1929	0.1819	5.69	0.4929	0.4478	9.16	0.9051	0.8026	11.32
	30	0.0306	0.0294	3.83	0.0397	0.0441	-11.02	0.1732	0.1773	-2.35	0.4518	0.4323	4.31	0.8290	0.7751	6.49
	40	0.0324	0.0300	7.32	0.0407	0.0420	-3.02	0.1734	0.1689	2.63	0.4475	0.4166	6.91	0.8064	0.7387	8.39
	50	0.0331	0.0298	9.91	0.0407	0.0414	-1.75	0.1719	0.1645	4.30	0.4400	0.4064	7.64	0.7971	0.7289	8.56
N(0.3, 0.3)	Epoch	0			0.1			0.3			0.5			0.7		
	10	0.1017	0.0596	41.41	0.0664	0.0476	28.36	0.0764	0.1048	-37.05	0.2171	0.2534	-16.71	0.4886	0.4714	3.52
	20	0.0956	0.0544	43.12	0.0620	0.0447	27.92	0.0770	0.0919	-19.28	0.2223	0.1870	15.87	0.4923	0.3102	36.98
	30	0.1013	0.0530	47.65	0.0662	0.0448	32.32	0.0739	0.0846	-14.50	0.2091	0.1690	19.15	0.4622	0.2988	35.35
	40	0.1218	0.0560	54.07	0.0806	0.0457	43.30	0.0712	0.0781	-9.75	0.1882	0.1717	8.77	0.4166	0.3322	20.26
N(0.5, 0.5)	50	0.1081	0.0492	54.52	0.0714	0.0433	39.34	0.0734	0.0797	-8.54	0.2007	0.1982	1.22	0.4480	0.4217	5.86
	Epoch	0			0.1			0.3			0.5			0.7		
	10	0.2409	0.1069	55.64	0.1747	0.0818	53.20	0.0966	0.0903	6.46	0.1134	0.1551	-36.79	0.2440	0.2541	-4.14
	20	0.2352	0.0764	67.53	0.1720	0.0643	62.62	0.0949	0.0826	12.94	0.1096	0.1269	-15.81	0.2319	0.1981	14.59
	30	0.2548	0.0721	71.69	0.1889	0.0624	66.99	0.1050	0.0824	21.56	0.1048	0.1162	-10.96	0.2069	0.2141	-3.47
N(0.7, 0.7)	40	0.2663	0.0870	67.35	0.2005	0.0786	60.81	0.1121	0.0889	20.65	0.1045	0.1111	-6.30	0.1894	0.2115	-11.66
	50	0.2609	0.0682	73.88	0.1957	0.0631	67.75	0.1127	0.0876	22.30	0.1059	0.1129	-6.53	0.1858	0.2167	-16.65
	Epoch	0			0.1			0.3			0.5			0.7		
	10	0.4242	0.1126	73.46	0.3365	0.0917	72.75	0.2018	0.1003	50.28	0.1309	0.1436	-9.72	0.1466	0.2044	-39.45
	20	0.3994	0.0968	75.77	0.3200	0.0892	72.12	0.1931	0.1113	42.35	0.1277	0.1221	4.42	0.1379	0.1500	-8.75
N(0.7, 0.7)	30	0.4051	0.1001	75.29	0.3291	0.1031	68.67	0.2105	0.1485	29.44	0.1422	0.1350	5.02	0.1358	0.1445	-6.40
	40	0.3603	0.0845	76.54	0.2974	0.0830	72.08	0.2012	0.1457	27.60	0.1478	0.1319	10.73	0.1401	0.1428	-1.97
	50	0.3104	0.0984	68.29	0.2568	0.1042	59.40	0.1841	0.1618	12.15	0.1448	0.1309	9.5891	0.1420	0.1418	0.14

图 11: 使用 $N(x, x)$ 噪音

表 1: 实验统计

Epoch	Methods		Improve(%)
	LSTM	JS	
10	0.4253	0.4440	-4.3903
20	0.4373	0.4730	-8.1526
30	0.3957	0.4166	-5.2926
40	0.3931	0.4221	-7.3938
50	0.3813	0.4264	-11.8368
60	0.3753	0.4019	-7.0887
70	0.3899	0.3797	2.6051
80	0.4325	0.3997	7.5715
90	0.4749	0.3623	23.7157
100	0.5363	0.4032	24.8283

5.5 气象预报

6 Appendix

$$\begin{aligned}
 MISE[\hat{f}(\cdot; h)] &:= \mathbb{E} \left[ISE[\hat{f}(\cdot; h)] \right] \\
 &= \mathbb{E} \left[\int (\hat{f}(x; h) - f(x))^2 dx \right] \\
 &= \int \mathbb{E} \left[(\hat{f}(x; h) - f(x))^2 \right] dx \\
 &= \int MSE[\hat{f}(x; h)] dx
 \end{aligned} \tag{13}$$

The squared integral of a function f

$$R(f) := \int f(x)^2 dx \tag{14}$$

The second moment of the kernel is denoted as

$$\mu_2(K) := \int z^2 K(z) dx \tag{15}$$

The convolution between real functions f and g , $f * g$

$$(f * g)(x) := \int f(x - y)g(y)dy = (g * f)(x) \tag{16}$$

$$K_h(x) = e^{-\frac{x^2}{2h^2}} \tag{17}$$

$$\mathbb{E}[\hat{f}(x; h)] = \frac{1}{n} \sum_{i=1}^n \mathbb{E}[K_h(x - X_i)] = \int K_h(x - y)f(y)dy \quad (18)$$

$$MISE(h) = E\left(\int (\hat{f}(x) - f(x))^2 dx\right) = \frac{R(K)}{nh} + \frac{1}{4}h^4 \mu_2^2(K)R(f'') + o(h^4 + (nh)^{-1}) \quad (19)$$

the optimal bandwidth that manimizes the MISE is as follows:

$$h_{MISE} = \left[\frac{R(K)}{\mu_2^2(K)R(f'')n} \right]^{1/5} \quad (20)$$

参考文献

- Emanuel Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065 – 1076, 1962. doi:10.1214/aoms/1177704472. URL <https://doi.org/10.1214/aoms/1177704472>.
- Simon J. Sheather. Density estimation. *Statistical Science*, 19(4):588–597, 2004. doi:10.1214/088342304000000297. URL <https://doi.org/10.1214/088342304000000297>.
- David W. Scott. *Multivariate Density Estimation*. Wiley Series in Probability and Statistics. 3 2015. ISBN 9780471697558. doi:10.1002/9781118575574.
- Mats Rudemo. Empirical choice of histograms and kernel density estimators. *Scandinavian Journal of Statistics*, 9(2):65–78, 1982. ISSN 03036898, 14679469. URL <http://www.jstor.org/stable/4615859>.
- Adrian W. Bowman. An alternative method of cross-validation for the smoothing of density estimates. *Biometrika*, 71(2):353–360, 1984. ISSN 00063444. URL <http://www.jstor.org/stable/2336252>.
- Charles J. Stone. An Asymptotically Optimal Window Selection Rule for Kernel Density Estimates. *The Annals of Statistics*, 12(4):1285 – 1297, 1984. doi:10.1214/aos/1176346792. URL <https://doi.org/10.1214/aos/1176346792>.
- David W. Scott and George R. Terrell. Biased and unbiased cross-validation in density estimation. *Journal of the American Statistical Association*, 82(400):1131–1146, 1987. doi:10.1080/01621459.1987.10478550. URL <https://www.tandfonline.com/doi/abs/10.1080/01621459.1987.10478550>.
- Michael Woodroofe. On Choosing a Delta-Sequence. *The Annals of Mathematical Statistics*, 41(5):1665 – 1671, 1970. doi:10.1214/aoms/1177696810. URL <https://doi.org/10.1214/aoms/1177696810>.
- S. J. Sheather and M. C. Jones. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53(3):683–690, 1991. ISSN 00359246. URL <http://www.jstor.org/stable/2345597>.
- S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1): 79–86, 1951. doi:10.1214/aoms/1177729694. URL <https://doi.org/10.1214/aoms/1177729694>.

- M. Lydia, S. Suresh Kumar, A. Immanuel Selvakumar, and G. Edwin Prem Kumar. A comprehensive review on wind turbine power curve modeling techniques. *Renewable and Sustainable Energy Reviews*, 30:452–460, 2014. ISSN 1364-0321. doi:<https://doi.org/10.1016/j.rser.2013.10.030>. URL <https://www.sciencedirect.com/science/article/pii/S1364032113007296>.
- A. Garcia, J.L. Torres, E. Prieto, and A. de Francisco. Fitting wind speed distributions: a case study. *Solar Energy*, 62(2):139–144, 1998. ISSN 0038-092X. doi:[https://doi.org/10.1016/S0038-092X\(97\)00116-3](https://doi.org/10.1016/S0038-092X(97)00116-3). URL <https://www.sciencedirect.com/science/article/pii/S0038092X97001163>.
- S.H. Jangamshetti and V.G. Ran. Optimum siting of wind turbine generators. *IEEE Transactions on Energy Conversion*, 16(1):8–13, 2001. doi:10.1109/60.911396.
- Zhilong Qin, Wenyuan Li, and Xiaofu Xiong. Estimating wind speed probability distribution using kernel density method. *Electric Power Systems Research*, 81(12):2139–2146, 2011. ISSN 0378-7796. doi:<https://doi.org/10.1016/j.epsr.2011.08.009>. URL <https://www.sciencedirect.com/science/article/pii/S0378779611001969>.
- Bo Hu, Yudun Li, Hejun Yang, and He Wang. Wind speed model based on kernel density estimation and its application in reliability assessment of generating systems. *Journal of Modern Power Systems and Clean Energy*, 5(2):220–227, 2017. doi:10.1007/s40565-015-0172-5.
- Jooyoung Jeon and James W. Taylor. Using conditional kernel density estimation for wind power density forecasting. *Journal of the American Statistical Association*, 107(497):66–79, 2012. doi:10.1080/01621459.2011.643745. URL <https://doi.org/10.1080/01621459.2011.643745>.