An RP-Interval Pearson-Fisher Chi-square Approach to Testing Goodness-of-fit for Location-scale Distributions

Jie Lee

Beijing Normal University - Hong Kong Baptist University United International College-Statistics

2021 年 10 月 11 日

Part1. Introduction

What is goodness-of-fit test

- Basically, GOF(goodness-of-fit test) is to test whether the data is following a specific distribution or a distribution family.
- Formally,we have:

$$H_0 : F \in \mathcal{P}_0, H_1 : F \notin \mathcal{P}_0 \tag{1}$$

Select a distribution $F^* \in \mathcal{P}_0$ such that:

$$m(F_n, F^*) = \min_{F \in \mathcal{P}_0} m(F_n, F) \tag{2}$$

Then calculate the exact distribution or approxiamte distribution of $m(F_n, F^*)$, and find the criteria of α (eg 0.05):$\xi(1-\alpha)$ so that

$$P(S(x_1, ..., x_n) \geq \xi(1-\alpha)|H_0) \leq \alpha \tag{3}$$

We want to test the null hypothesis

$$H_0 : F(x) \sim N(\mu, \sigma^2) \tag{4}$$

against the alternative hypothesis that $F(x)$ is not a normal distribution, where μ and σ are unknown mean and standard deviation parameters of the normal distribution $N(\mu, \sigma^2)$.
Classical test for normality: SW test, KS test, X^2-test,...

For the computation of χ^2-test, the sample data should be grouped into m connective intervals, then the difference between observed and expected frequencies in each interval is calculated in order to obtain the χ^2 test statistics.

Pearson's χ^2-statistic (Pearson, 1933) is defined by

$$\chi^2 = \sum_{i=1}^{k} \frac{(n_i - np_i)^2}{np_i}. \tag{5}$$

The classical χ^2-statistic for testing hypothesis (1) is constructed as follow. Classify the domain of the underlying distribution into a sufficiently large number k intervals artificially like

$$I_1 = (-\infty, a_1), I_2 = [a_1, a_2), ..., I_{k-1} = [a_{k-2}), a_{k-1}), I_k = [a_{k-1}, +\infty)] \tag{6}$$

where the classification points $a_1, ..., a_{k-1}$ are somewhat arbitrarily fixed. Denote n_i as the number of sample points that are locate at the interval I_i,and

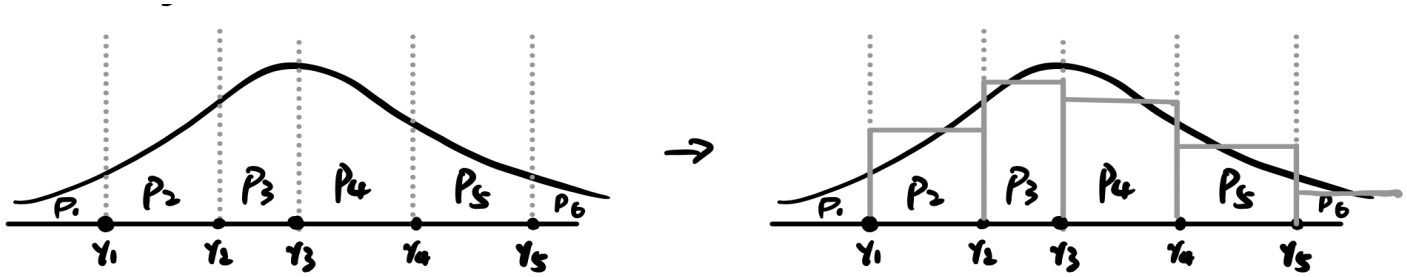
$$p_i = \int_{I_i} dF(x) \qquad for \qquad i = 1...k \tag{7}$$

Apply Chi-test for continous distribution

How to change the continuous distribution into discrete distribution?

- Equal size
- Equal probability
- Use MSE Representative points

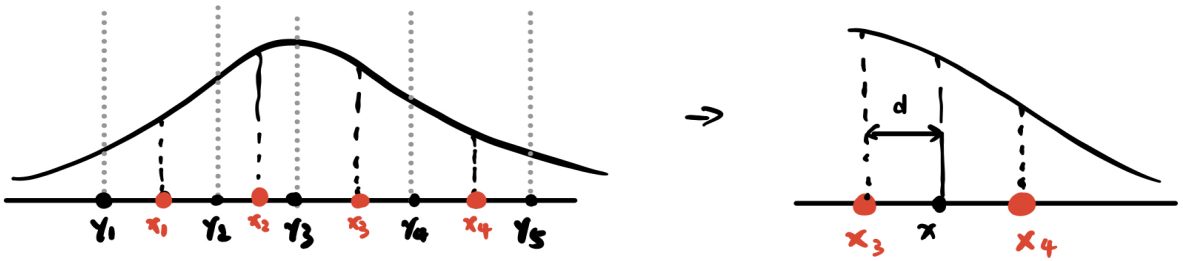
Brief intro of MSE RPs



$$\text{MSE-Loss} = \int_{-\infty}^{\infty} \min_{1 \le i \le m} \left(\frac{x_i - x}{\sigma} \right)^2 \cdot \phi(x)\, dx$$

$$\text{or} = \frac{1}{\sigma^2} \int_{-\infty}^{\infty} \min_{1 \le i \le m} (x_i - x)^2 \cdot \phi(x)\, dx$$

$$= \frac{1}{\sigma^2} E_\phi \left[\min_{1 \le i \le m} (x_i - x)^2 \right]$$



Example: Normal Case

As the hypothesis is $N(\mu, \sigma^2)$, for a random sample we need to use MLE to estimate the two unkonwn parameters μ and σ.

Accroding to Fisher, the statistic here should be follow a Chi-squared distribution which degree of freedom equal to $k - s - 1 \quad = \quad k - 3$, that is :

$$x^2 \xrightarrow{D} x^2(k-3), \quad n \to \infty \tag{8}$$

Let's m be the number of RPs, if $m = 2l$ is an even number, then we only ask that $0 < x_1 < x_2 < ... < x_l$; When m is odd number $m = 2l + 1$, we only ask that $0 = x_0 < x_1 < ... < x_l$. Then the MSE-RP lost function will be equal to:

$$\begin{aligned}\frac{1}{2} f(x_1, \cdots, x_l) = \int_0^{\frac{1}{2}(x_1+x_2)} (x - x_1)^2 \varphi(x) dx + \int_{\frac{1}{2}(x_1+x_2)}^{\frac{1}{2}(x_2+x_3)} (x - x_2)^2 \varphi(x) dx + \cdots \\ + \int_{\frac{1}{2}(x_{l-1}+x_1)}^{\infty} (x - x_l)^2 \varphi(x) dx\end{aligned} \tag{9}$$

and

$$\begin{aligned}\frac{1}{2} f(x_0, x_1, \cdots, x_l) = \int_0^{\frac{1}{2}x_1} x^2 \varphi(x) dx + \int_{\frac{1}{2}x_1}^{\frac{1}{2}(x_1+x_2)} (x - x_1)^2 \varphi(x) dx + \cdots \\ + \int_{\frac{1}{2}(x_{l-1}+x_l)}^{\infty} (x - x_l)^2 \varphi(x) dx\end{aligned} \tag{10}$$

By calculating the partial derivative of $x_1, ..., x_m$ and let them equal to zero we get:

$$\begin{cases} \varphi(0) - \varphi\left(\frac{1}{2}(x_1 + x_2)\right) = x_1\left[\Phi\left(\frac{1}{2}(x_1 + x_2)\right) - \Phi(0)\right] \\ \varphi\left(\frac{1}{2}(x_1 + x_2)\right) - \varphi\left(\frac{1}{2}(x_2 + x_3)\right) = x_2\left[\Phi\left(\frac{1}{2}(x_2 + x_3)\right) - \Phi\left(\frac{1}{2}(x_1 + x_2)\right)\right] \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \varphi\left(\frac{1}{2}(x_{1-2} + x_{1-1})\right) - \varphi\left(\frac{1}{2}(x_{l-1} + x_1)\right) = x_{l-1}\left[\Phi\left(\frac{1}{2}(x_{l-1} + x_l)\right) - \Phi\left(\frac{1}{2}(x_{l-1} + x_{1-2})\right)\right] \\ \varphi\left(\frac{1}{2}(x_{l-1} + x_l)\right) = x_l\left[1 - \Phi\left(\frac{1}{2}(x_{l-1} + x_1)\right)\right] \end{cases} \tag{11}$$

and

$$\begin{cases} \varphi\left(\frac{1}{2}x_1\right) - \varphi\left(\frac{1}{2}(x_1 + x_2)\right) = x_1\left[\Phi\left(\frac{1}{2}(x_1 + x_2)\right) - \Phi\left(\frac{1}{2}x_1\right)\right] \\ \varphi\left(\frac{1}{2}(x_1 + x_2)\right) - \varphi\left(\frac{1}{2}(x_2 + x_3)\right) = x_2\left[\Phi\left(\frac{1}{2}(x_2 + x_3)\right) - \Phi\left(\frac{1}{2}(x_1 + x_2)\right)\right] \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \varphi\left(\frac{1}{2}(x_{l-2} + x_{l-1})\right) - \varphi\left(\frac{1}{2}(x_{l-1} + x_l)\right) = x_{l-1}\left[\Phi\left(\frac{1}{2}(x_{l-1} + x_l)\right) - \Phi\left(\frac{1}{2}(x_{l-2} + x_{l-1})\right)\right] \\ \varphi\left(\frac{1}{2}(x_{l-1} + x_l)\right) = x_l\left[1 - \Phi\left(\frac{1}{2}(x_{l-1} + x_l)\right)\right] \end{cases} \tag{12}$$

Step 1. Set $x_1 = \frac{1}{l+1} x_{2(l-1),1}$ as initial value, when $l = 2$, $x_{21} = 0.79778$ is known, let $LP = 0$, $RP = x_{2(l-1),1}$, then we know that the solution $x_{2l,1}$ must in the interval of (LP, RP)

Step 2. As the initial value of x_1 and the first l equations is given then we calculate the $x_2 = g_2(x_1), ..., x_l = g_l(x_l)$,from the last equation we calculate the x_1^* which $g_i(x)$ is the function in (5) that we could use x_{i-1} to calculate the value x_i.

Step 3. As we set the error as the accruacy of the calculation, we may face to these two situation: a) if $|x_l - x_l^*| < \epsilon$,then $x_1, ..., x_l$ is the solution of equation set (5). b) if $x_l < x_l^* + \epsilon$,then set $LP = x_1, x_1 = \frac{1}{2}(LP + RP)$,then back to Step.2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
m=2	0.7978845608																	
m=3	1.2240063619																	
m=4	0.4527800346	1.51041761																
m=5	0.7645675711	1.72414741																
m=6	0.3177163693	1.00010605	1.893594812															
m=7	0.5605768883	1.18814695	2.033369181															
m=8	0.2450941790	0.75600528	1.343909279	2.151945705														
m=9	0.4436386476	0.91879588	1.476391739	2.254663636														
m=10	0.1996228516	0.60985751	1.057825045	1.591340442	2.345095886													
m=11	0.3674581121	0.75236729	1.178826407	1.692639034	2.425746394													
m=12	0.1684377467	0.51184650	0.876779522	1.285711288	1.783029896	2.498435263												
m=13	0.3137732303	0.63825073	0.986949078	1.381262661	1.864518208	2.564525349												
m=14	0.1457062441	0.44132089	0.750442618	1.085635313	1.467528268	1.938612407	2.625062508											
m=15	0.2738568549	0.55476369	0.851134412	1.174879178	1.546056754	2.006474396	2.680866119											
m=16	0.1283950298	0.38804830	0.656759119	0.942340456	1.256231197	1.618046386	2.069017226	2.73258957										
m=17	0.2429942181	0.49088231	0.749287085	1.025596680	1.330895862	1.684442439	2.126970785	2.78076191										
m=18	0.1147688484	0.34634551	0.584302017	0.833861698	1.102100179	1.399826687	1.746003090	2.18092726	2.82581726									
m=19	0.2184088412	0.44035547	0.669796847	0.911666214	1.172801431	1.463790839	1.803345278	2.23137342	2.86811605									
m=20	0.1037625818	0.31279138	0.526488152	0.748533289	0.983642445	1.238467214	1.523414292	1.85697739	2.27871394	2.90796068								
m=21	0.1983570608	0.39935556	0.605892180	0.821441699	1.050551899	1.299724709	1.579214159	1.90732306	2.32328876	2.94560721								
m=22	0.0946859479	0.28519859	0.479232054	0.679485196	0.889275753	1.113018753	1.357093171	1.63162230	1.95473882	2.36538600	2.98127431							
m=23	0.1816876915	0.36540026	0.553325367	0.748011750	0.952654299	1.171560767	1.411007127	1.68100291	1.99952743	2.40525191	3.01515014							
m=24	0.0870718117	0.26210145	0.439853533	0.622370365	0.812084114	1.012091530	1.226611986	1.46183369	1.72766579	2.04194802	2.44309852	3.04739769						
m=25	0.1676096369	0.33680595	0.509283450	0.686972116	0.872211732	1.068019165	1.278539718	1.50988568	1.77187657	2.08222404	2.47910960	3.07815909						
m=26	0.0805926905	0.24247998	0.406516546	0.574287630	0.747634678	0.928822913	1.120803072	1.32765743	1.55543171	1.81386461	2.12054942	2.51344547	3.10755894					
m=27	0.1555610842	0.31238876	0.471823393	0.635364436	0.804782315	0.982281084	1.170755905	1.37423466	1.59870409	1.85382927	2.15709351	2.54624674	3.13570709					
m=28	0.0750121607	0.22560202	0.377918780	0.533219051	0.692933750	0.858775420	1.032897187	1.21814686	1.41850484	1.63990497	1.89194485	2.19200500	2.57763739	3.16270084				
m=29	0.1451319316	0.29129116	0.439556835	0.591118869	0.747352583	0.909922659	1.080939246	1.26320932	1.46067132	1.67921131	1.92836457	2.22541515	2.60772729	3.18862677				
m=30	0.0701552458	0.21092809	0.353109629	0.497713705	0.645876292	0.798926980	0.958490355	1.12663986	1.30614684	1.50091233	1.71677886	1.96322380	2.25744032	2.63661420	3.21356224			
m=31	0.1360157510	0.27287628	0.411463697	0.552739196	0.697794549	0.847921236	1.004709848	1.17020212	1.34713801	1.53938492	1.75274537	1.99664270	2.28818417	2.66438550	3.23757665			
m=32	0.0658896598	0.19805183	0.331378306	0.466699523	0.604933624	0.747135704	0.894565117	1.04878332	1.21180438	1.38634034	1.57622808	1.78723322	2.02872840	2.31773940	2.69111958	3.26073249		
m=33	0.1279789364	0.25666100	0.386776751	0.519114478	0.654556709	0.794127754	0.939059594	1.09088846	1.25160405	1.42389338	1.61156545	1.82035161	2.05957677	2.34618923	2.71688700	3.28308620		
m=34	0.0621135291	0.18666118	0.312182310	0.439366049	0.568965886	0.701834557	0.838970259	0.98158142	1.13118223	1.28974075	1.45992136	1.64550745	1.85219838	2.08927396	2.37360862	2.74175153	3.30468893	
m=35	0.1208402611	0.24227202	0.364907313	0.489401119	0.616477896	0.746965198	0.881838860	1.02228686	1.16980396	1.32633884	1.49453530	1.67815306	1.88286147	2.11789761	2.40006534	2.76577096	3.32558718	
m=36	0.0587471237	0.17651252	0.295100182	0.415088537	0.537104128	0.661846936	0.790123032	0.92288893	1.06131469	1.20687784	1.36150957	1.52783482	1.70959134	1.91242024	2.14551797	2.42562083	2.78899781	3.34582334

Part2. χ^2-test base on MSE RPs

MSE RPs for Logistic distribution

$$\begin{cases} \varphi(0) - \varphi\left(\frac{1}{2}(x_1+x_2)\right) = x_1\left[\Phi\left(\frac{1}{2}(x_1+x_2)\right) - \Phi(0)\right] \\ \varphi\left(\frac{1}{2}(x_1+x_2)\right) - \varphi\left(\frac{1}{2}(x_2+x_3)\right) = x_2\left[\Phi\left(\frac{1}{2}(x_2+x_3)\right) - \Phi\left(\frac{1}{2}(x_1+x_2)\right)\right] \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \varphi\left(\frac{1}{2}(x_{1-2}+x_{1-1})\right) - \varphi\left(\frac{1}{2}(x_{l-1}+x_1)\right) = x_{l-1}\left[\Phi\left(\frac{1}{2}(x_{l-1}+x_l)\right) - \Phi\left(\frac{1}{2}(x_{l-1}+x_{1-2})\right)\right] \\ \varphi\left(\frac{1}{2}(x_{l-1}+x_l)\right) = x_l\left[1 - \Phi\left(\frac{1}{2}(x_{l-1}+x_1)\right)\right] \end{cases} \tag{13}$$

For the logistic distribution, the $p.d.f$ is

$$f(x) = \frac{e^{-x}}{(1+e^{-x})^2} \tag{14}$$

The $c.d.f$ is

$$F(x) = \frac{1}{1+e^{-x}} \tag{15}$$

RPs of logistic distribution

Table 3.2: MSE representative point of Logistic distribution $m = [2, 30]$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
m = 2	1.386294														
m = 3	2.289109														
m = 4	0.828806	2.971407													
m = 5	1.441413	3.522743													
m = 6	0.593142	1.935422	3.986331												
m = 7	1.063273	2.352669	4.386679												
m = 8	0.462204	1.457016	2.715349	4.739209											
m = 9	0.844945	1.798062	3.036863	5.054269											
m = 10	0.378741	1.174069	2.100135	3.325994	5.339073										
m = 11	0.701885	1.464318	2.371981	3.588916	5.599										
m = 12	0.320846	0.985263	1.724873	2.619542	3.830112	5.838035									
m = 13	0.600618	1.23866	1.961876	2.847095	4.053007	6.059358									
m = 14	0.278327	0.849698	1.468539	2.179641	3.057813	4.260237	6.265418								
m = 15	0.525061	1.07491	1.679384	2.381321	3.254122	4.453871	6.458134								
m = 16	0.245768	0.747384	1.280975	1.874461	2.569329	3.437968	4.635643	6.639209							
m = 17	0.46648	0.950242	1.471282	2.056201	2.74552	3.610875	4.806904	6.809894							
m = 18	0.220034	0.66731	1.137187	1.648364	2.226499	2.911403	3.77414	4.968861	6.971418						
m = 19	0.419711	0.851957	1.310844	1.814145	2.386831	3.068172	3.928789	5.122445	7.124636						
m = 20	0.199182	0.602878	1.023158	1.473214	1.970144	2.5384	3.216835	4.07572	5.268507	7.270406					
m = 21	0.381495	0.772375	1.182986	1.625853	2.117567	2.682181	3.35822	4.215673	5.407738	7.409385					
m = 22	0.181942	0.54988	0.930357	1.333056	1.770003	2.257403	2.818994	3.49304	4.349301	5.540765	7.542206				
m = 23	0.349677	0.706565	1.07849	1.474642	1.906667	2.390465	2.949526	3.621901	4.47716	5.668122	7.669392				
m = 24	0.16745	0.505505	0.853274	1.218092	1.608774	2.036668	2.517435	3.074361	3.745324	4.599738	5.790275	7.791407			
m = 25	0.322773	0.651203	0.991366	1.350221	1.736292	2.160692	2.638885	3.193997	3.863756	4.717447	5.907617	7.908617			
m = 26	0.155097	0.467795	0.788174	1.121932	1.475743	1.857897	2.279319	2.755316	3.308875	3.977604	4.830677	6.020535	8.021428		
m = 27	0.299719	0.603963	0.91754	1.245861	1.59537	1.974173	2.393042	2.867151	3.419373	4.087216	4.939755	6.129345	8.130148		
m = 28	0.14445	0.435346	0.73243	1.040215	1.363885	1.709702	2.08562	2.502284	2.974762	3.525828	4.192902	5.044979	6.234336	8.235061	
m = 29	0.279745	0.563166	0.854137	1.156953	1.476617	1.819244	2.192663	2.607415	3.078475	3.628535	4.29494	5.146616	6.335772	8.336433	
m = 30	0.135157	0.407125	0.68414	0.969852	1.268377	1.584571	1.924425	2.295669	2.708752	3.178572	3.727751	4.39357	5.24489	6.433863	8.434449

MSE RPs for Laplace distribution

$$\begin{cases} \varphi(0) - \varphi\left(\frac{1}{2}(x_1+x_2)\right) = x_1\left[\Phi\left(\frac{1}{2}(x_1+x_2)\right) - \Phi(0)\right] \\ \varphi\left(\frac{1}{2}(x_1+x_2)\right) - \varphi\left(\frac{1}{2}(x_2+x_3)\right) = x_2\left[\Phi\left(\frac{1}{2}(x_2+x_3)\right) - \Phi\left(\frac{1}{2}(x_1+x_2)\right)\right] \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \varphi\left(\frac{1}{2}(x_{1-2}+x_{1-1})\right) - \varphi\left(\frac{1}{2}(x_{l-1}+x_1)\right) = x_{l-1}\left[\Phi\left(\frac{1}{2}(x_{l-1}+x_l)\right) - \Phi\left(\frac{1}{2}(x_{l-1}+x_{1-2})\right)\right] \\ \varphi\left(\frac{1}{2}(x_{l-1}+x_l)\right) = x_l\left[1 - \Phi\left(\frac{1}{2}(x_{l-1}+x_1)\right)\right] \end{cases} \tag{16}$$

For the laplace distribution, the $p.d.f$ is

$$f(x) = exp(-\frac{|x|}{b}) \tag{17}$$

The $c.d.f$ is

$$F(x) = \frac{1}{2} + \frac{1}{2}\text{sgn}(x)(1 - exp(-|x-\mu|)) \tag{18}$$

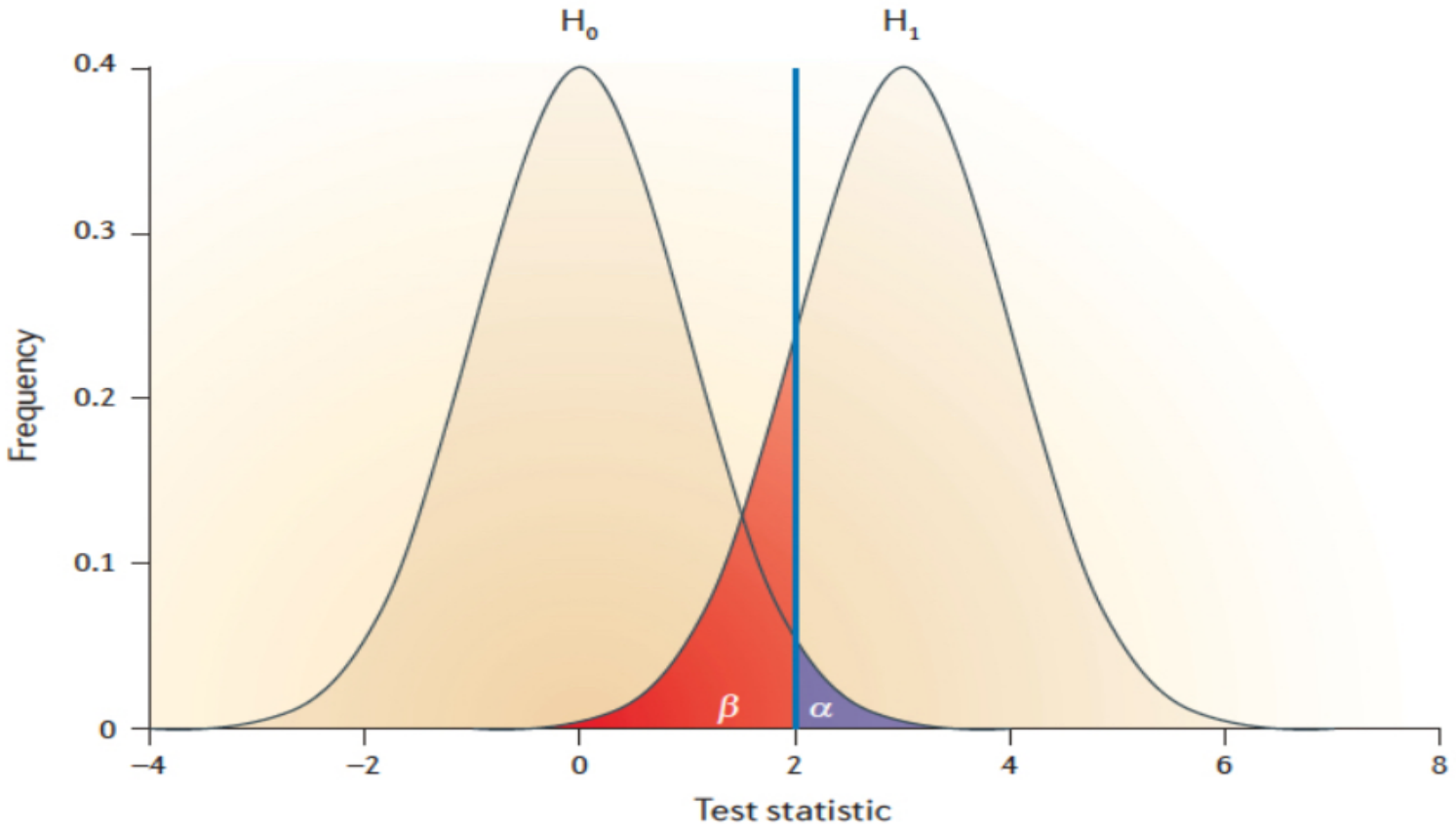
RPs of laplace distribution

Table 3.3: MSE representative point of Laplace distribution $m = [2, 30]$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
m = 2	1														
m = 3	2														
m = 4	0.593625	2.593623													
m = 5	1.18725	3.187247													
m = 6	0.423954	1.611202	3.611202												
m = 7	0.847907	2.035156	4.035156												
m = 8	0.330079	1.177986	2.365234	4.365234											
m = 9	0.660158	1.508065	2.695313	4.695313											
m = 10	0.270351	0.930508	1.778415	2.965664	4.965664										
m = 11	0.540702	1.200859	2.048766	3.236015	5.236015										
m = 12	0.228969	0.769671	1.429828	2.277735	3.464983	5.464983									
m = 13	0.457938	0.99864	1.658797	2.506704	3.693952	5.693952									
m = 14	0.198593	0.656531	1.197233	1.85739	2.705298	3.892546	5.892547								
m = 15	0.397187	0.855125	1.395826	2.055984	2.903891	4.091139	6.09114								
m = 16	0.175343	0.572529	1.030467	1.571169	2.231326	3.079233	4.266482	6.266482							
m = 17	0.350685	0.747872	1.20581	1.746511	2.406669	3.254576	4.441825	6.441825							
m = 18	0.156971	0.507656	0.904843	1.362781	1.903483	2.56364	3.411547	4.598796	6.598796						
m = 19	0.313942	0.664627	1.061814	1.519752	2.060454	2.720611	3.568518	4.755767	6.755767						
m = 20	0.142087	0.456029	0.806715	1.203901	1.661839	2.202541	2.862698	3.710606	4.897854	6.897854					
m = 21	0.284175	0.598117	0.948802	1.345988	1.803927	2.344628	3.004786	3.852693	5.039941	7.039942					
m = 22	0.129783	0.413958	0.7279	1.078585	1.475772	1.93371	2.474412	3.134569	3.982476	5.169725	7.169725				
m = 23	0.259567	0.543741	0.857684	1.208369	1.605555	2.063493	2.604195	3.264353	4.11226	5.299508	7.299509				
m = 24	0.119442	0.379009	0.663183	0.977125	1.327811	1.724997	2.182935	2.723637	3.383794	4.231701	5.418949	7.418949			
m = 25	0.238884	0.498451	0.782625	1.096567	1.447253	1.844439	2.302377	2.843079	3.503236	4.351143	5.538391	7.538391			
m = 26	0.110628	0.349512	0.609078	0.893253	1.207195	1.55788	1.955067	2.413005	2.953707	3.613864	4.461771	5.64902	7.649019		
m = 27	0.221255	0.460139	0.719706	1.003881	1.317823	1.668508	2.065695	2.523633	3.064334	3.724492	4.572399	5.759647	7.759647		
m = 28	0.103026	0.324281	0.563165	0.822732	1.106906	1.420848	1.771534	2.16872	2.626658	3.16736	3.827517	4.675424	5.862673	7.862672	
m = 29	0.206051	0.427306	0.66619	0.925757	1.209932	1.523874	1.874559	2.271746	2.729684	3.270385	3.930543	4.77845	5.965698	7.965698	
m = 30	0.096401	0.302452	0.523708	0.762592	1.022159	1.306333	1.620275	1.970961	2.368147	2.826085	3.366786	4.026944	4.874851	6.062099	8.062098

Part3. Monte Carlo Simulation

Type I and II Error



Simulation of Type I Error

Algorithm 1: Monte Carlo algorithm For Type I error

Input: input parameters α, k, n, N

Output: Type I error:α

Set $i = 1$;
while $i \leq N$ **do**
  Generate n random number $x_1, ..., x_n$from a Normal distribution;
  Use MLEs to estimate μ and σ;
  Apply transformation $y_i = \frac{x_i - \hat{\mu}}{\hat{\sigma}}$ for $i = 1, ..., n$;
  Use MSE-RP to calculate m_i and p_i;
  Calculate the statistic x^2;
  if $x^2 >= x^2_{\alpha,k-3,m}$ **then**
    $I_i = 1$;
  else
    $I_i = 0$;
  end
end
Calculate the **Power** $\alpha = \frac{\sum_{i=1}^{N} I_i}{N}$

Simulation of Power

Algorithm 2: Monte Carlo algorithm For Type II error

Input: input parameters α, k, n, N

Output: Type II error:β

Set $i = 1$;
while $i \leq N$ **do**
  Generate n random number $x_1, ..., x_n$from a Non-Normal distribution;
  Use MLEs to estimate μ and σ;
  Apply transformation $y_i = \frac{x_i - \hat{\mu}}{\hat{\sigma}}$ for $i = 1, ..., n$;
  Use MSE-RP to calculate m_i and p_i;
  Calculate the statistic x^2;
  if $x^2 >= x^2_{\alpha,k-3,m}$ **then**
    $I_i = 1$;
  else
    $I_i = 0$;
  end
end
Calculate the **Power** $\beta = \frac{\sum_{i=1}^{N} I_i}{N}$

Testing Normal distribution

Simulation of Power

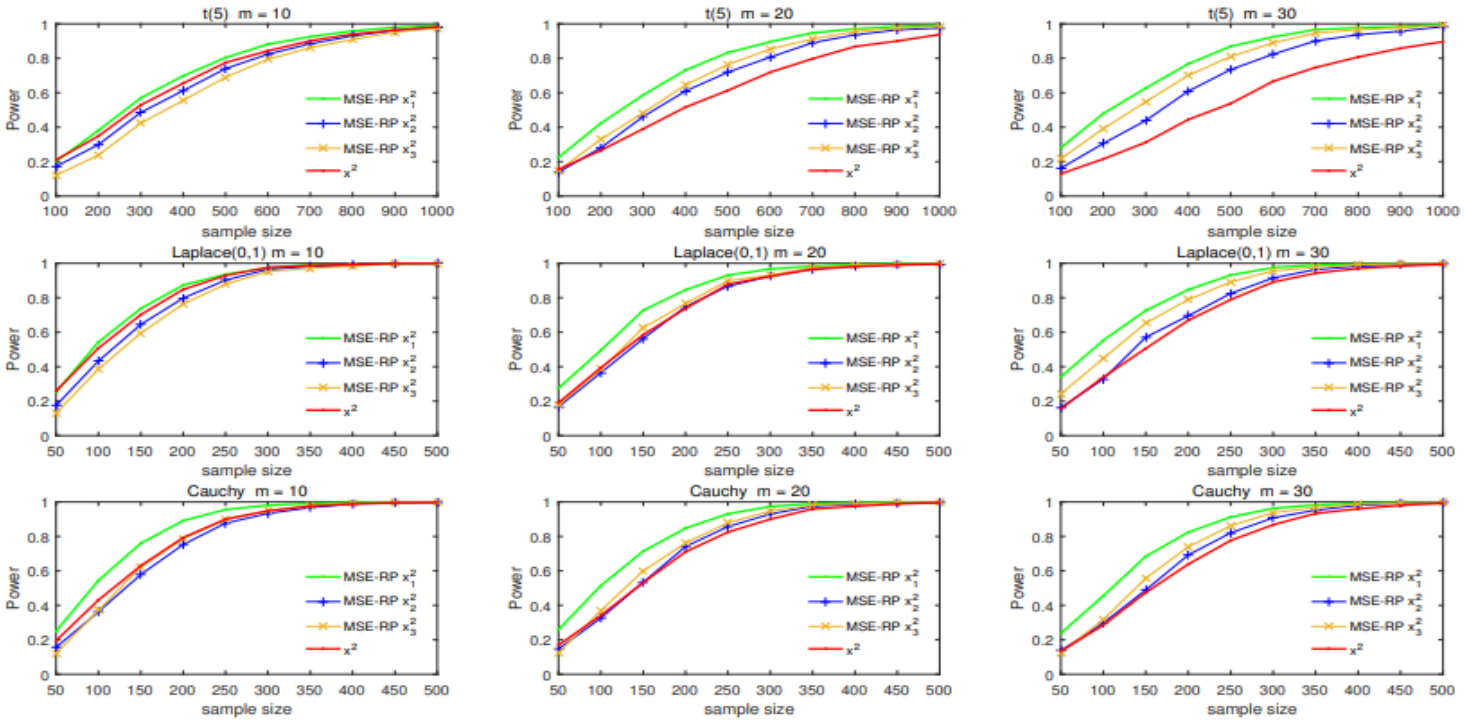


Figure 1. An illustration of power comparison based on symmetric fat-tailed distributions

Simulation of Power

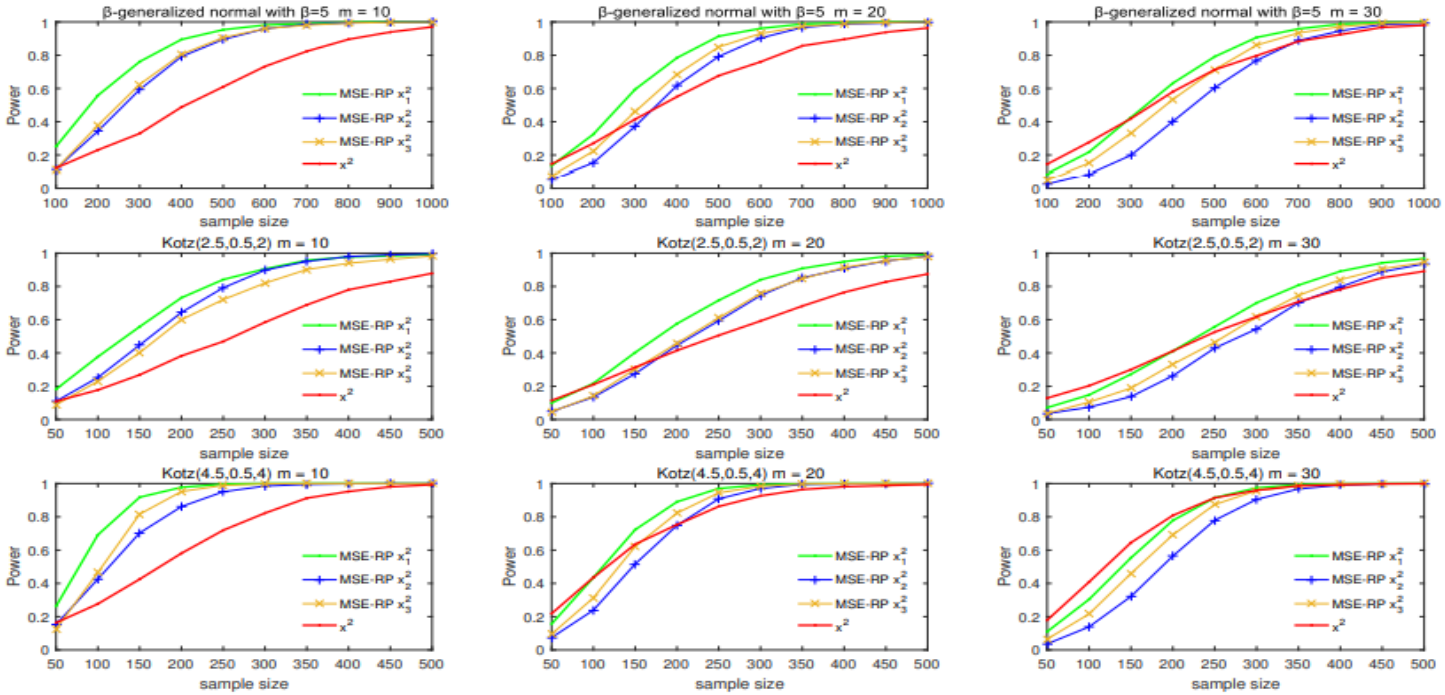


Figure 2. An illustration of power comparison based on symmetric thin-tailed distributions

Simulation of Power

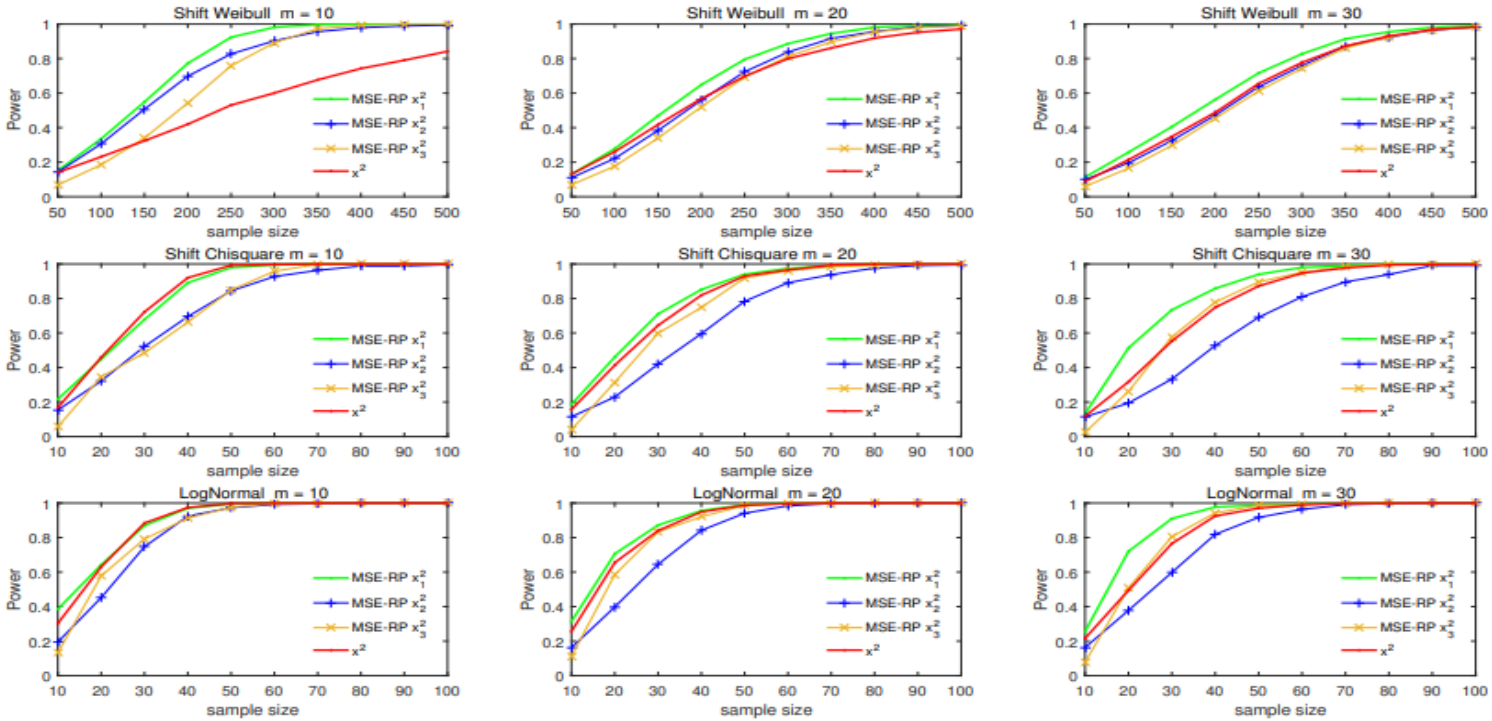
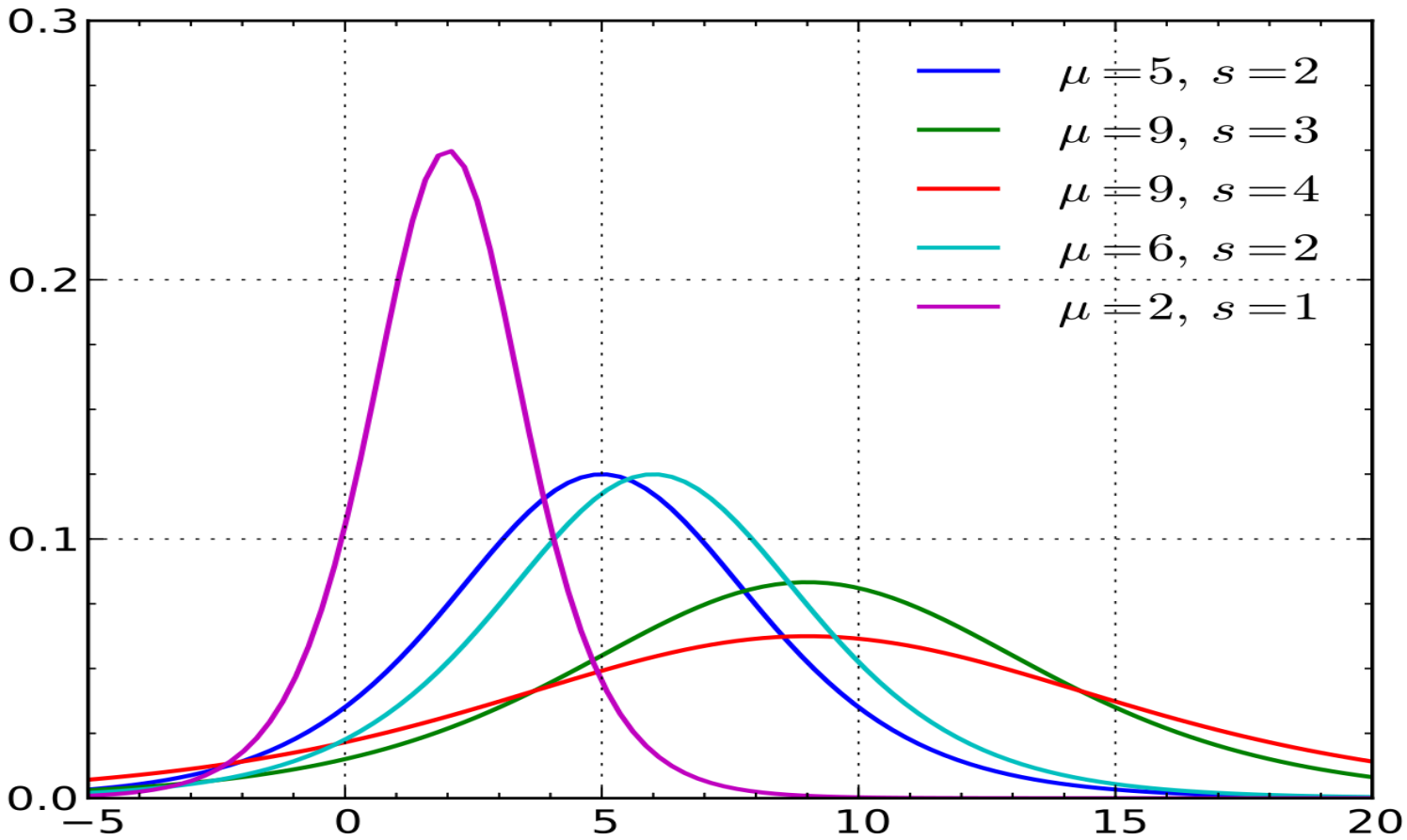


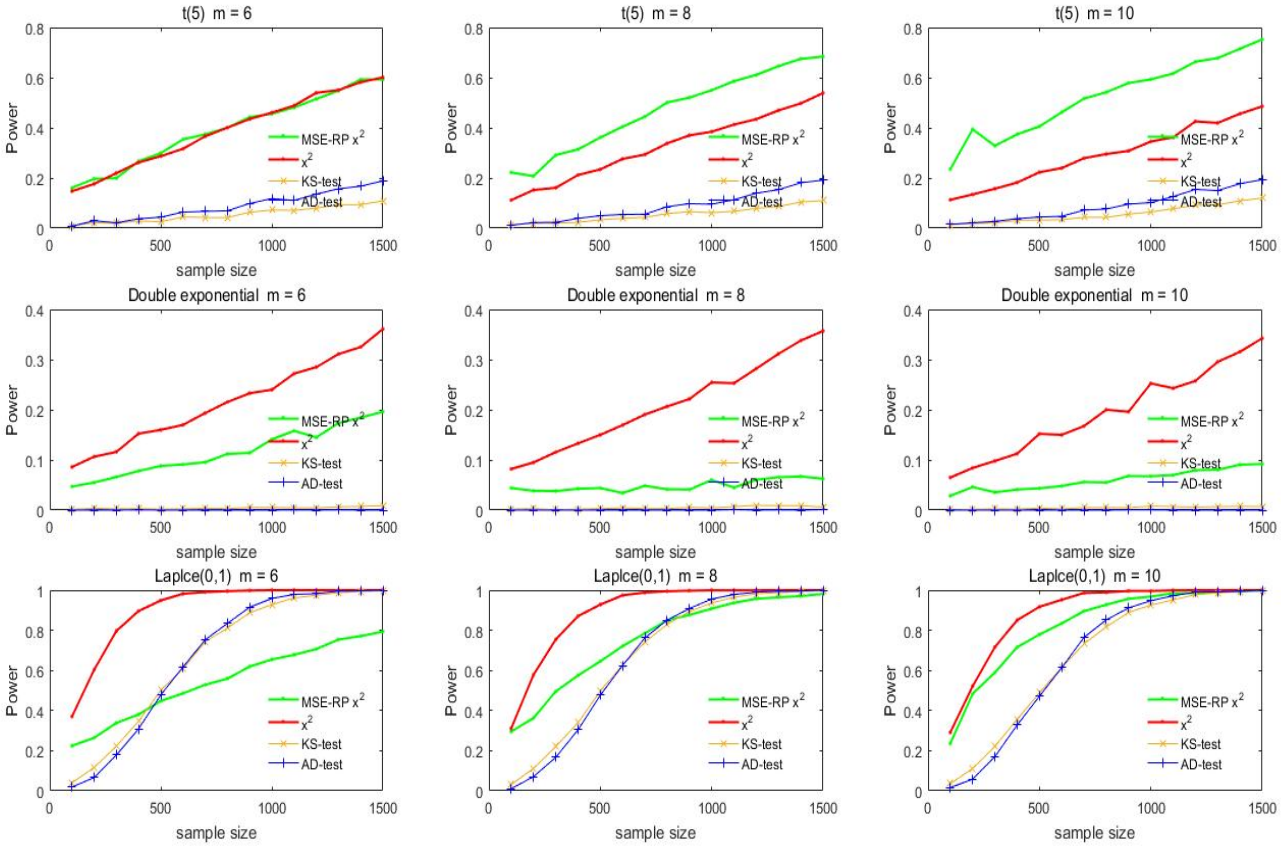
Figure 3. An illustration of power comparison based on asymmetric distributions

Testing Logistic distribution

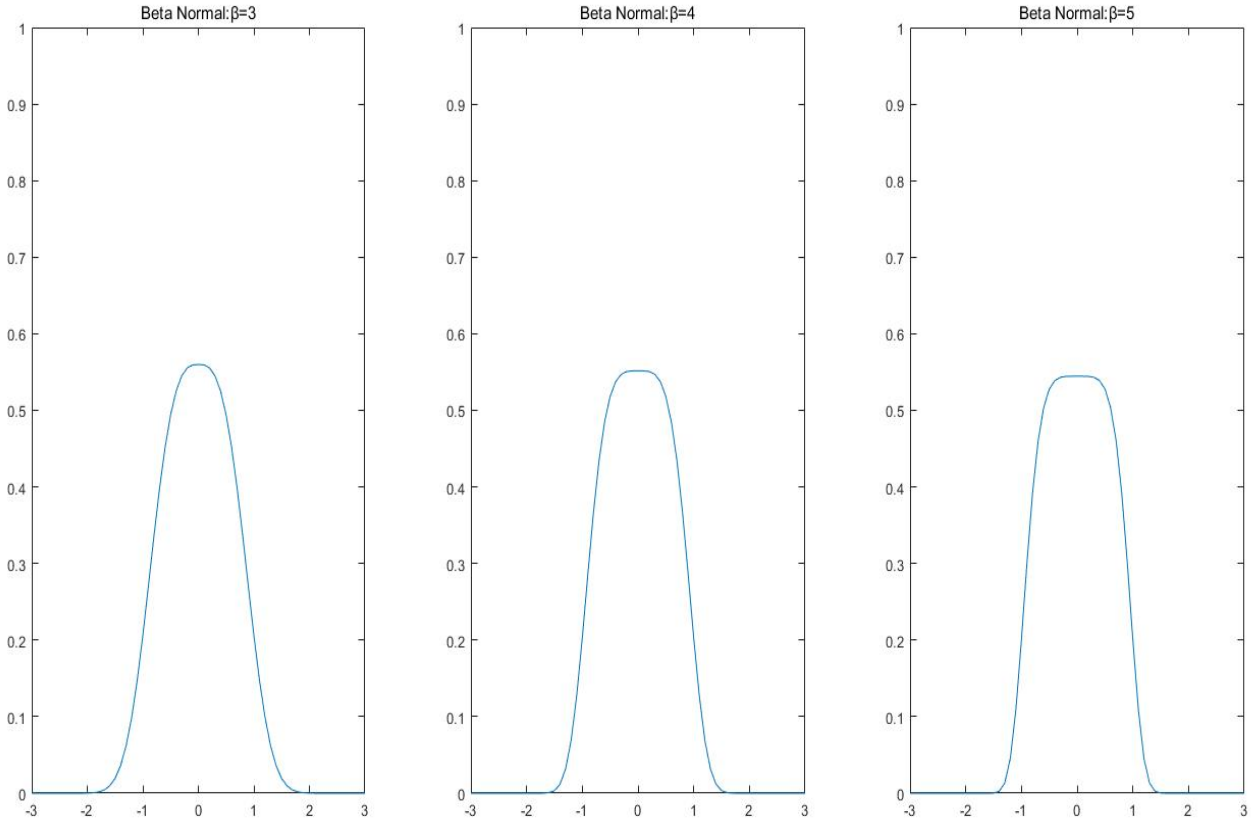
Testing Logistic distribution



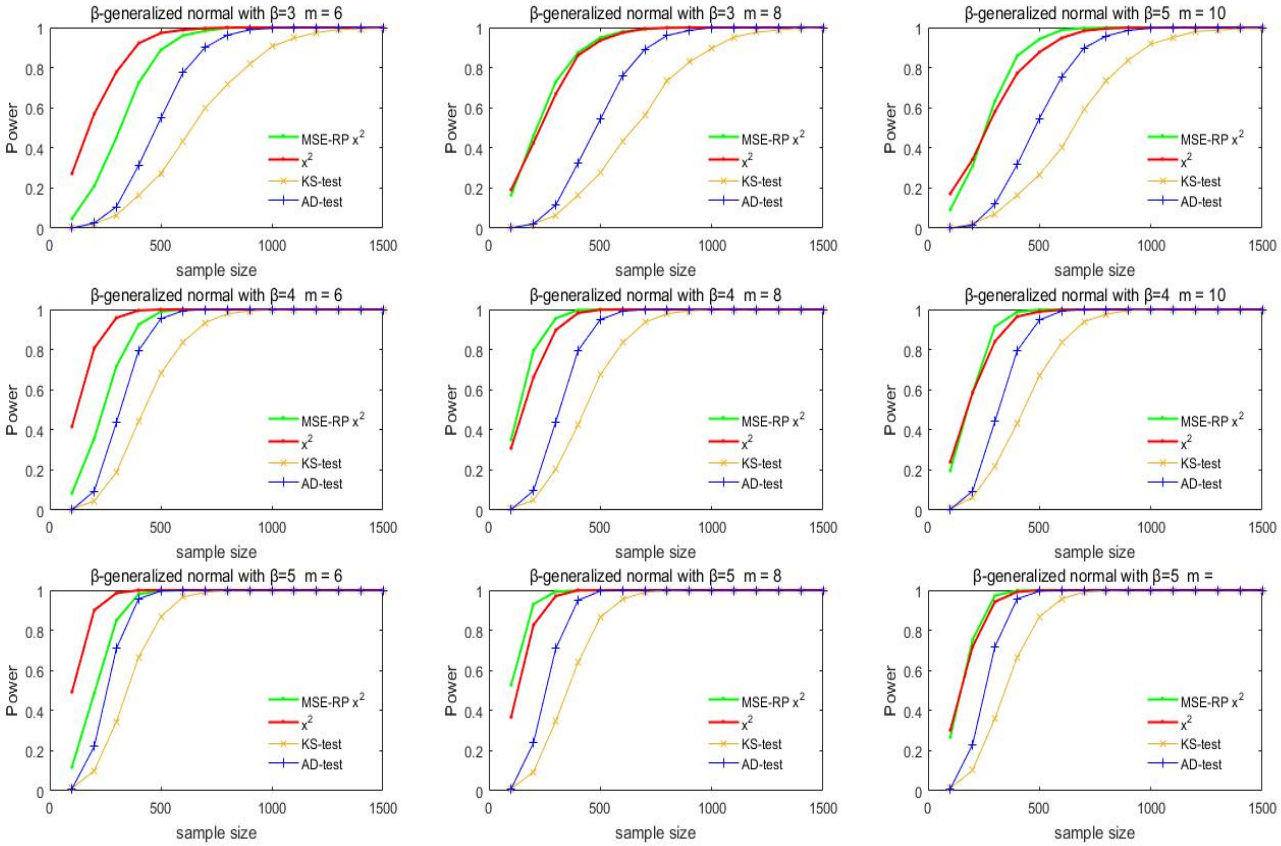
Symmetric fat tail



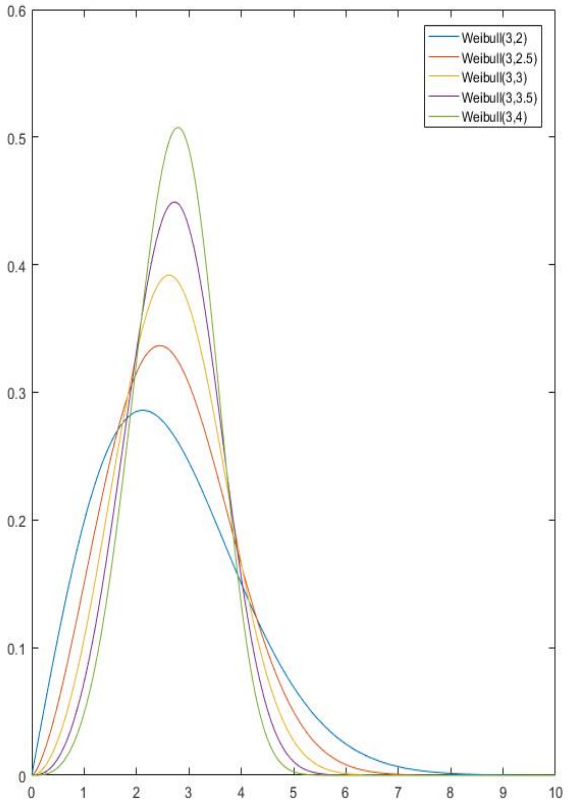
Symmetric thin tail

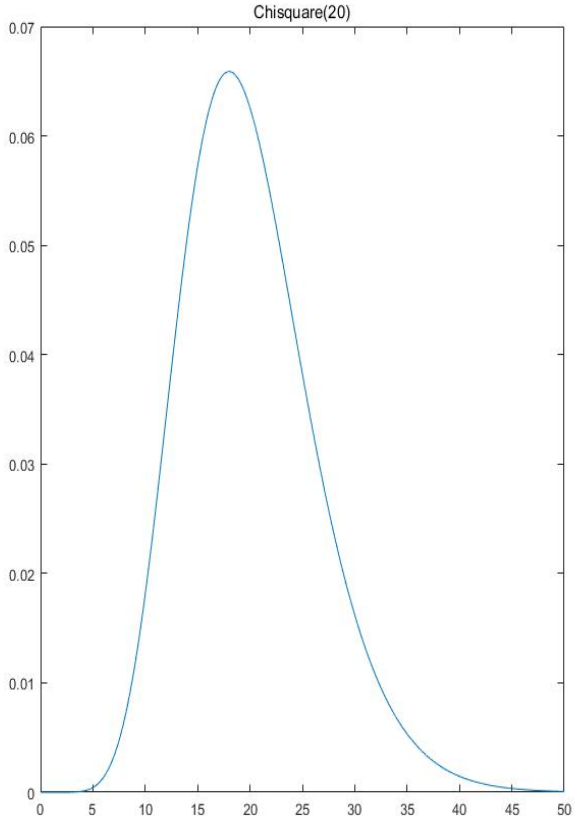


Symmetric thin tail

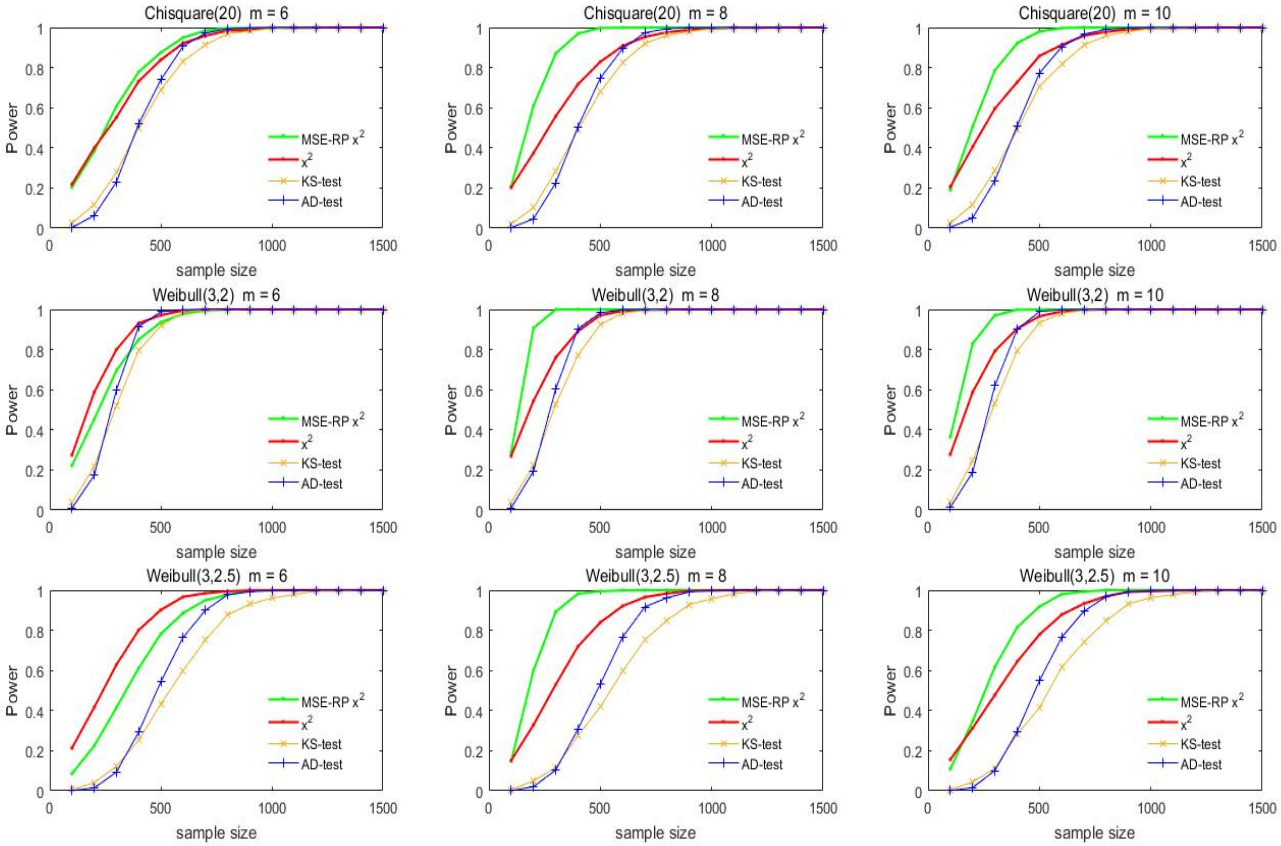


Asymmetric distribution

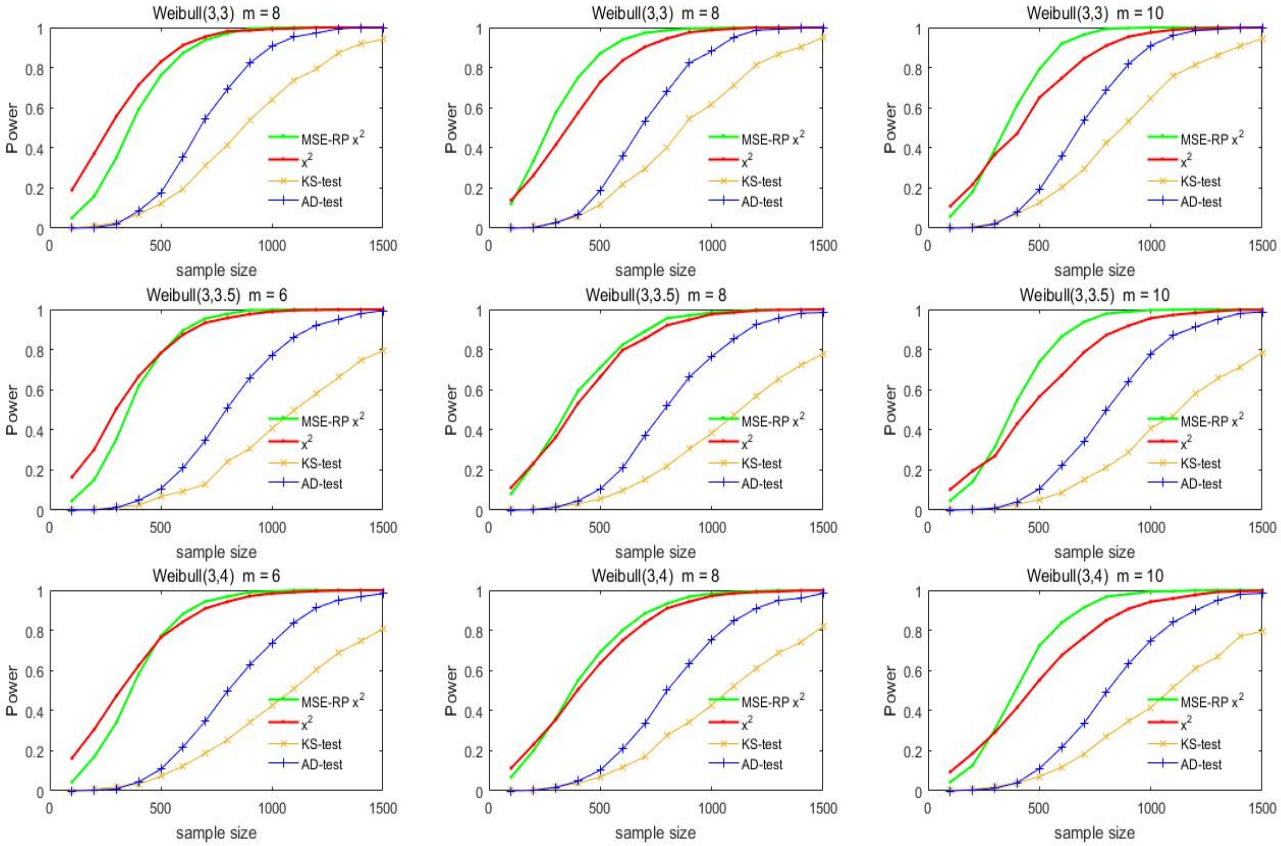




Asymmetric distribution

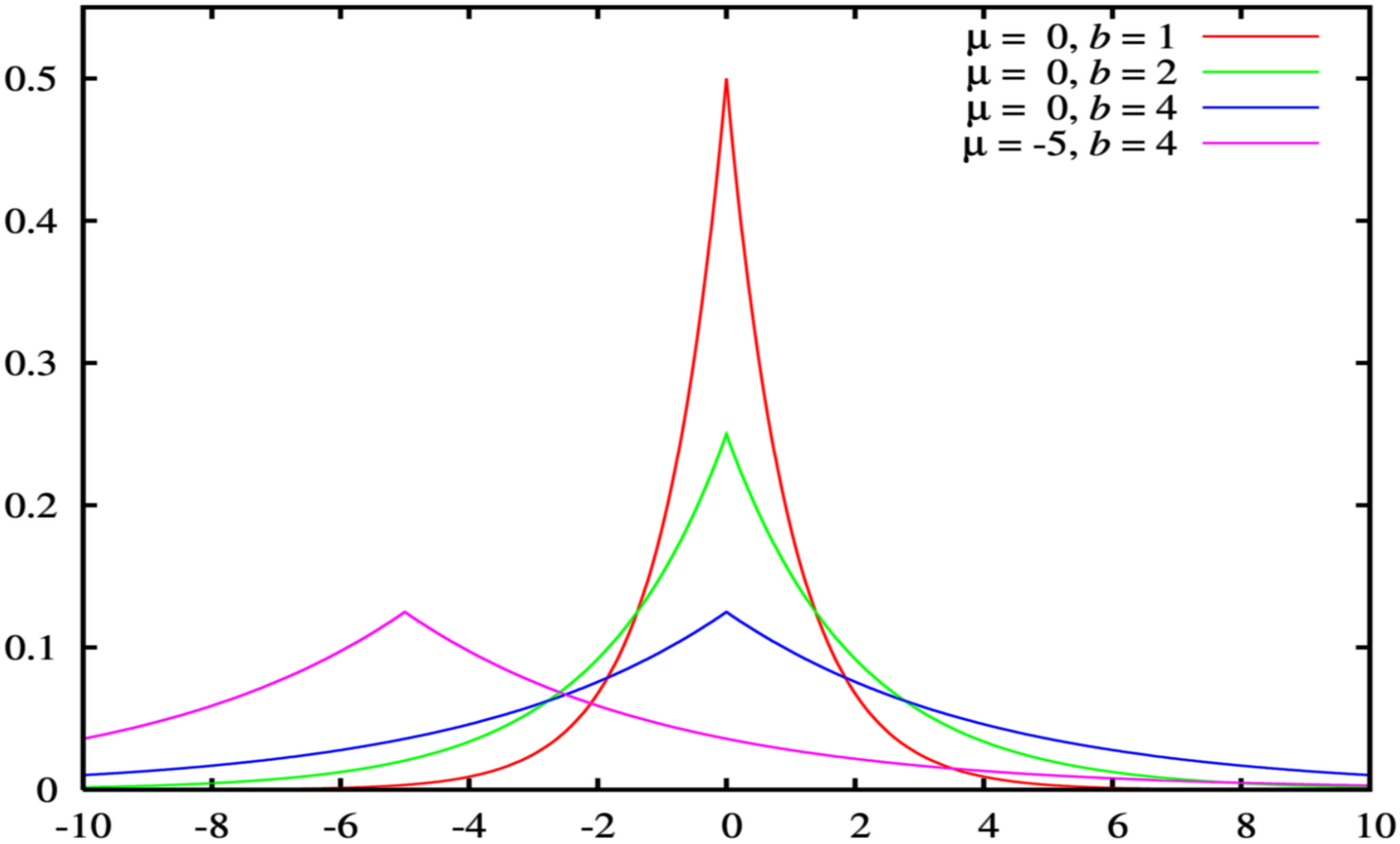


Asymmetric distribution

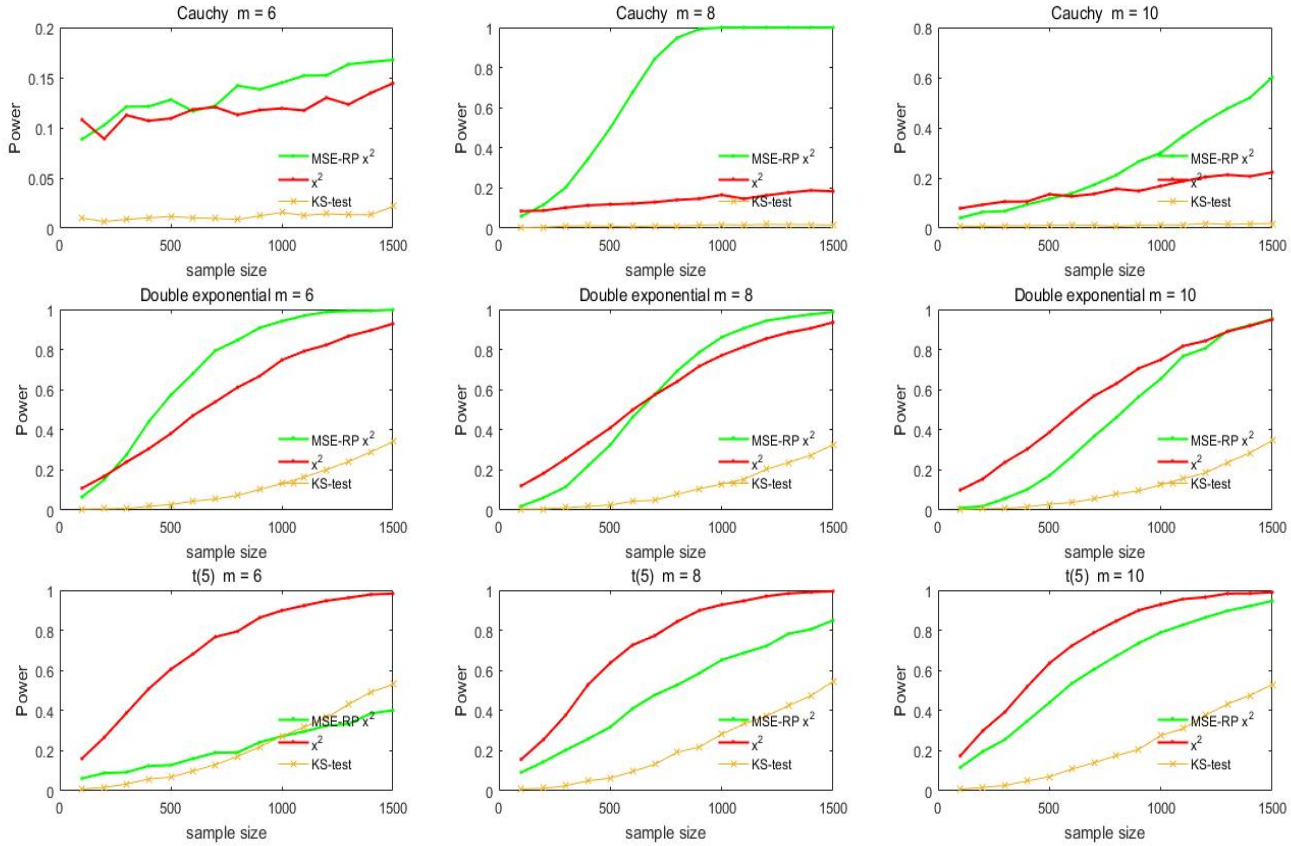


Testing Laplace distribution

Testing Laplace distribution

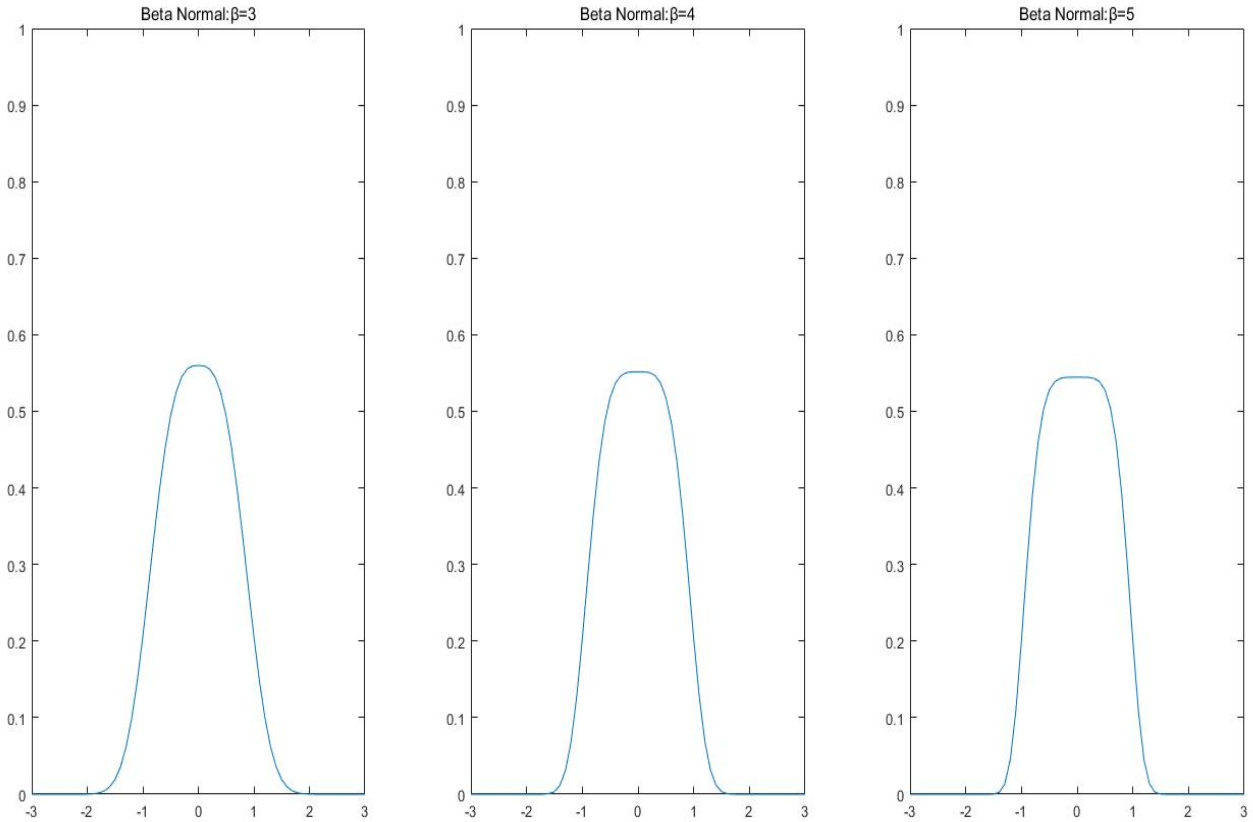


Symmetric fat tail

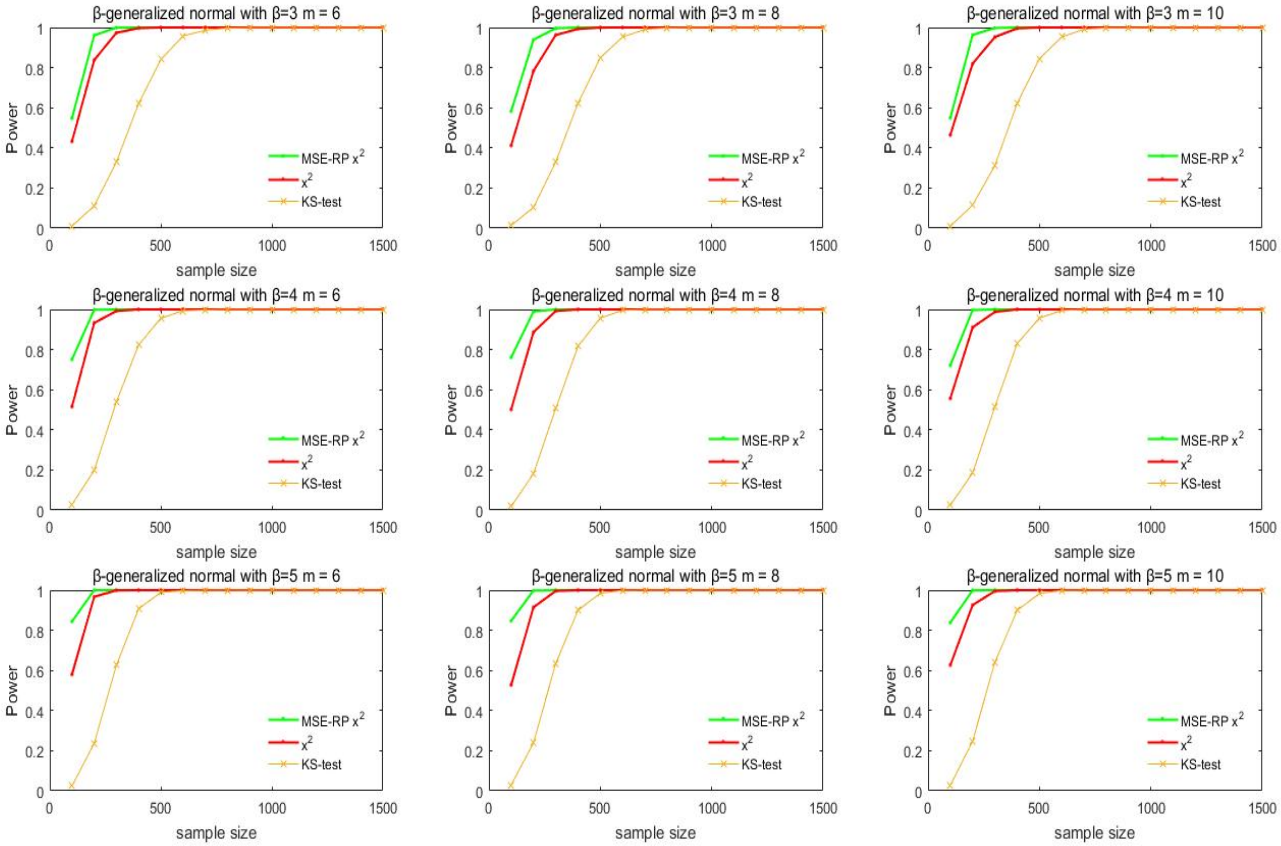


Symmetric thin tail

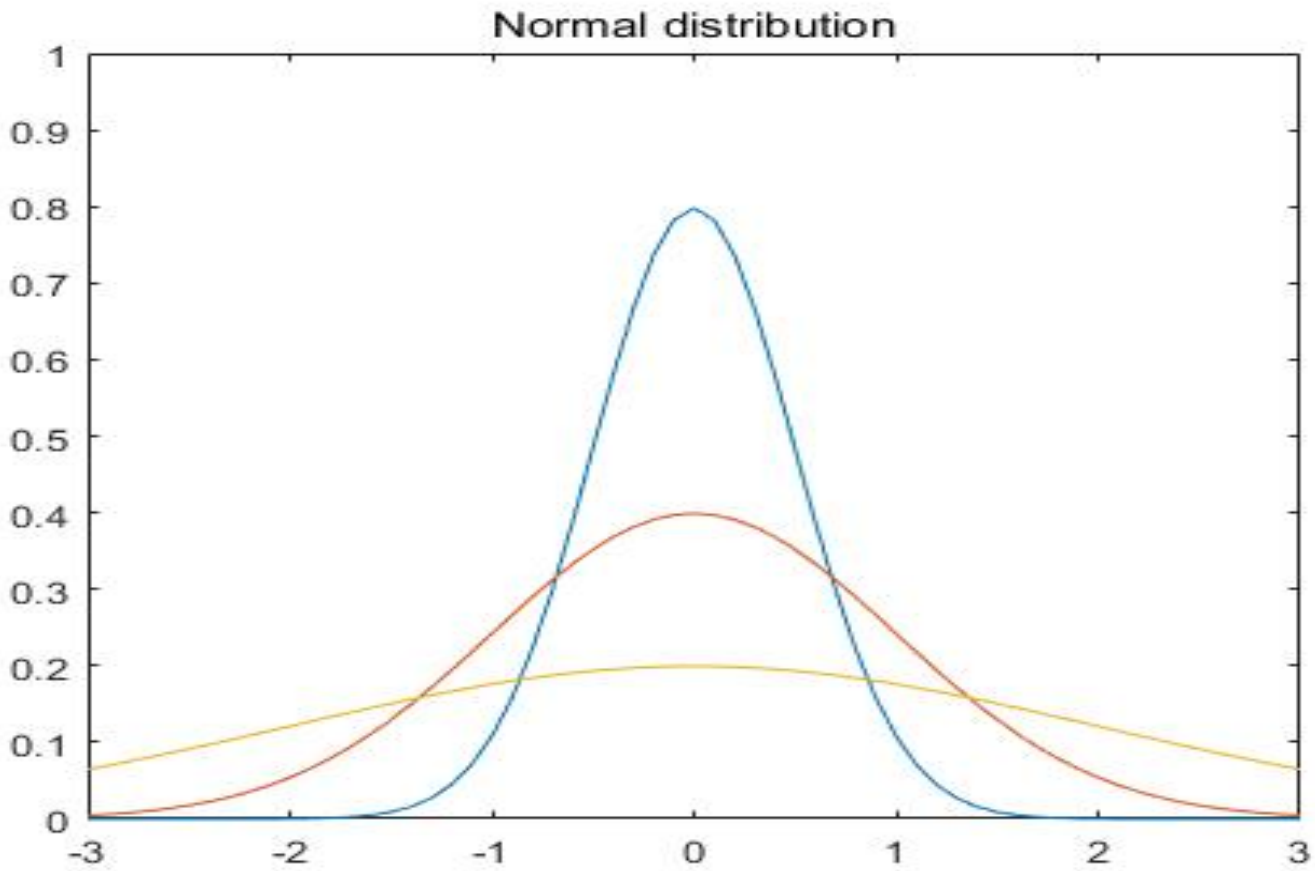
Beta Normal distribution with $\beta = 3, 4, 5$



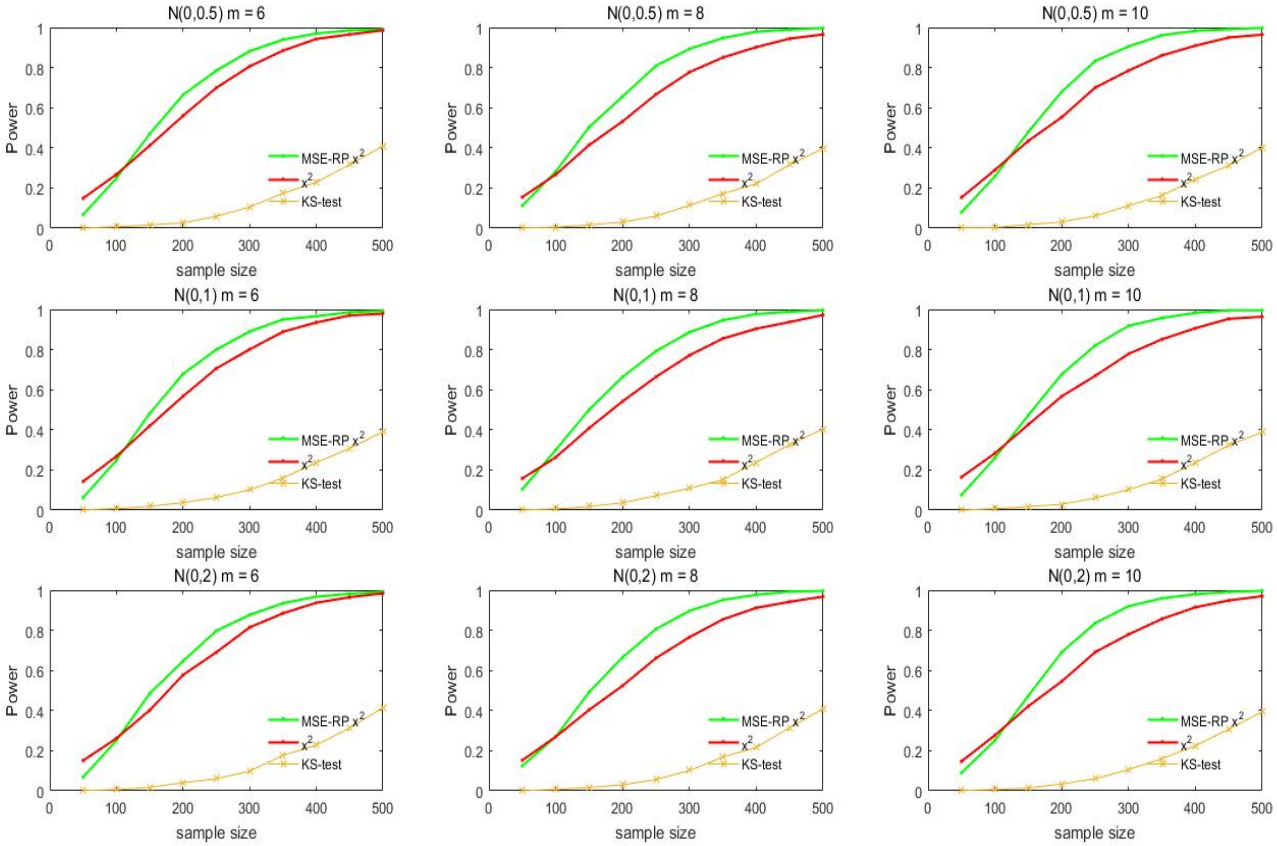
Symmetric thin tail



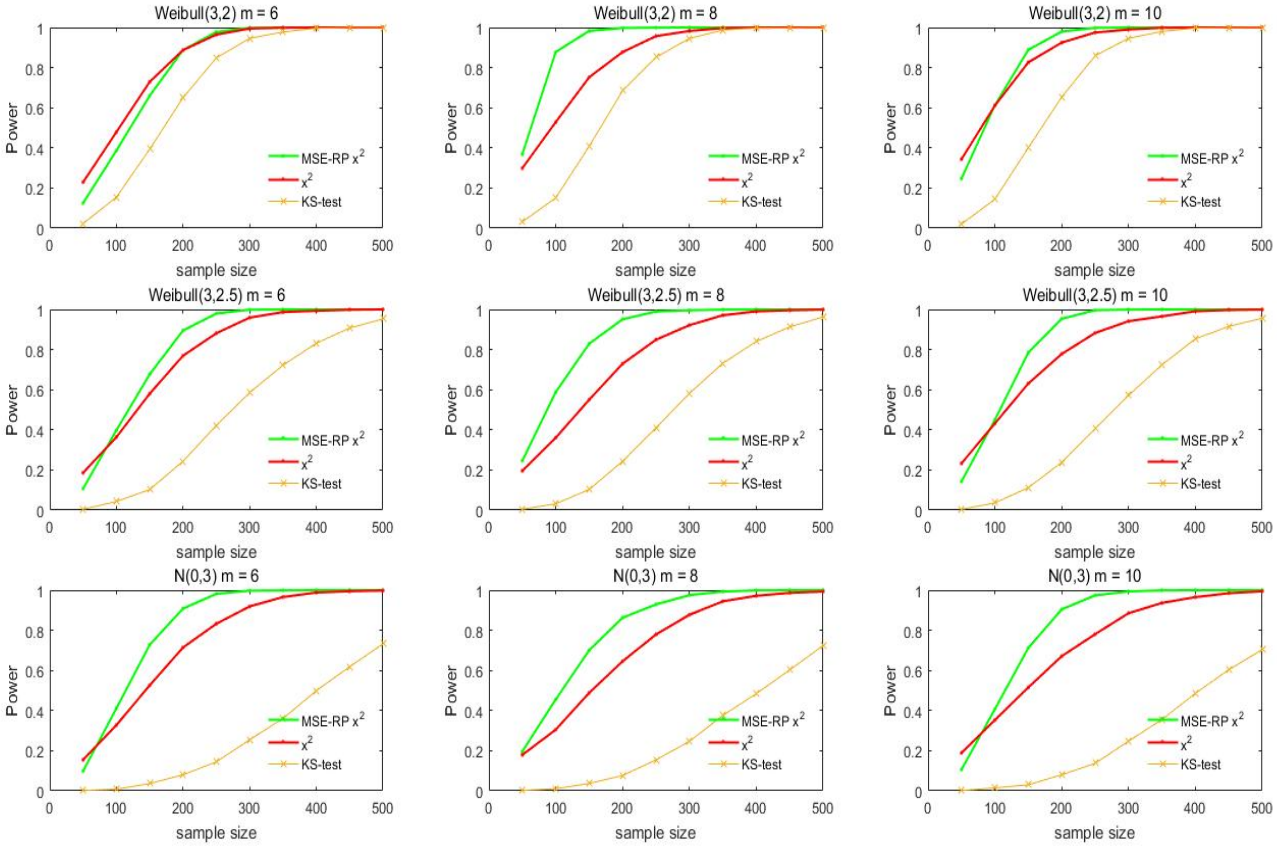
Symmetric thin tail



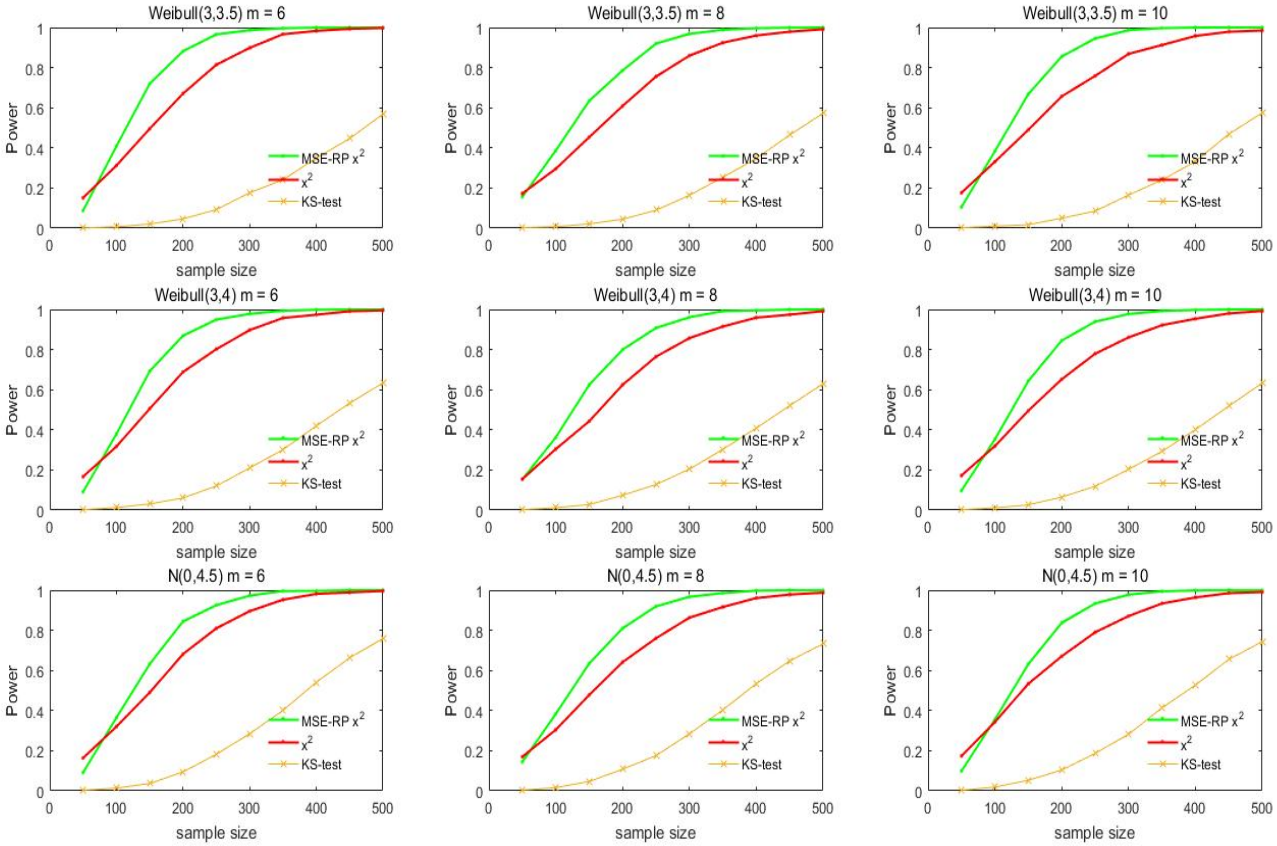
Symmetric thin tail



Asymmetric distribution



Asymmetric distribution

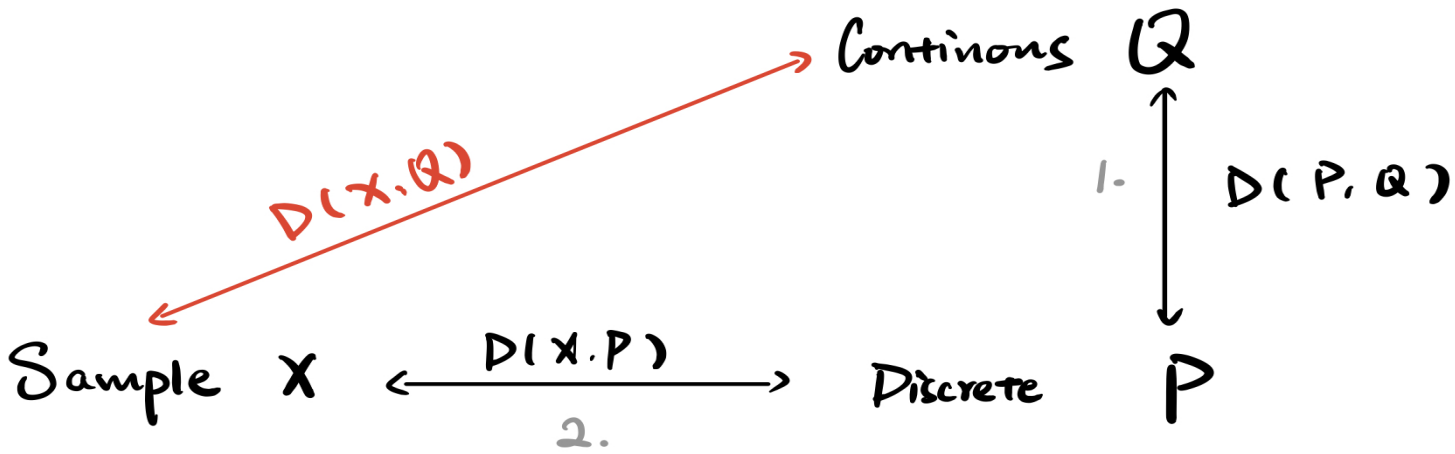


Part4. Discussion

What thing cause the power different?

1. number of representative points.
2. Alternative distribution (data).

Possible reason why the power is not stable



Q1: Can we directly use the MSE-RP to do the goodness-of-fit?Use kernel method to estimate the probabilistic density of sample then calculate MSE distance?

Q2: What about other measuring method? Before goodness-of-fit test, we need define what is 'distance' between distribution or data.

Part5. Distance in Statistic and Machine learning

As we need to compare two distributions P and Q, the distance can consturct in two form:

- $\frac{P}{Q}$: f-divergence such as KL divergence, JS divergence,W-divergence

- $P - Q$: Integral Probability Metric(IPM) such as Maximum Mean Discrepancy(MMD), Kernel Stein Discrepancy(KSD)...

f-divergency

- General form of f-divergence: $D_f(P||Q) = \int q(x) f(\frac{p(x)}{q(x)}) dx$

距离名称	计算公式	对应的f
总变差	$\frac{1}{2}\int \lvert p(x)-q(x)\rvert dx$	$\frac{1}{2}\lvert u-1\rvert$
KL散度	$\int p(x)\log\frac{p(x)}{q(x)}dx$	$u\log u$
逆KL散度	$\int q(x)\log\frac{q(x)}{p(x)}dx$	$-\log u$
Pearson χ^2	$\int \frac{(q(x)-p(x))^2}{p(x)}dx$	$\frac{(1-u)^2}{u}$
Neyman χ^2	$\int \frac{(p(x)-q(x))^2}{q(x)}dx$	$(u-1)^2$
Hellinger距离	$\int \left(\sqrt{p(x)}-\sqrt{q(x)}\right)^2 dx$	$(\sqrt{u}-1)^2$
Jeffrey距离	$\int (p(x)-q(x))\log\left(\frac{p(x)}{q(x)}\right)dx$	$(u-1)\log u$
JS散度	$\frac{1}{2}\int p(x)\log\frac{2p(x)}{p(x)+q(x)} + q(x)\log\frac{2q(x)}{p(x)+q(x)}dx$	$-\frac{u+1}{2}\log\frac{1+u}{2}+\frac{u}{2}\log u$

Three features of function f

- Recall the general form of f-divergence:

$$D_f(P||Q) = \int q(x) f(\frac{p(x)}{q(x)}) dx \tag{19}$$

1. f is a mapping from R^* to R
2. $f(1) = 0$
3. f is a convex function

- For the 3, we if a function is convex then $E[f(x)] \geq f(E(x))$, we have:
$D_f(P||Q) = \int q(x) f(\frac{p(x)}{q(x)}) dx = E[f(\frac{p(x)}{q(x)})] \geq f(E[\frac{p(x))}{q(x)}]) \geq$
$f(\int q(x) \frac{p(x)}{q(x)} dx) = f(1) = 0$

How to calculate f-divergence?

- Recall the general form of f-divergence:

$$D_f(P||Q) = \int q(x) f(\frac{p(x)}{q(x)}) dx \tag{20}$$

Need to know P and Q to calcualte, it's impossible in Machine Leanring and Deep Learning problem. We need to find the Conjugate function $f^*(y)$ for $f(x)$:

$$f^*(y) = \max_{x \in domf} (y^T x - f(x)) \tag{21}$$

$$D_f(P||Q) = \int q(x) f(\frac{p(x)}{q(x)}) dx \tag{22}$$

$$f^*(y) = \max_{x \in domf} (y^T x - f(x)) \tag{23}$$

- Let's conjuate function of $f(\frac{p(x)}{q(x)})$ is g:

$$f(\frac{p(x)}{q(x)}) = \max_{x \in domf} (\frac{p(x)}{q(x)} t - g(t)) \tag{24}$$

$$f(\frac{p(x)}{q(x)}) = \max_{x \in domf} (\frac{p(x)}{q(x)} t - g(t)) \tag{25}$$

$$D_f(P||Q) = \max_{t \in domg} \int [p(x)t - q(x)g(t)]dx \tag{26}$$

$$= \max_{t \in domg} (E_{x \sim p(x)}[t] - E_{x \sim q(x)}[g(t)]) \tag{27}$$

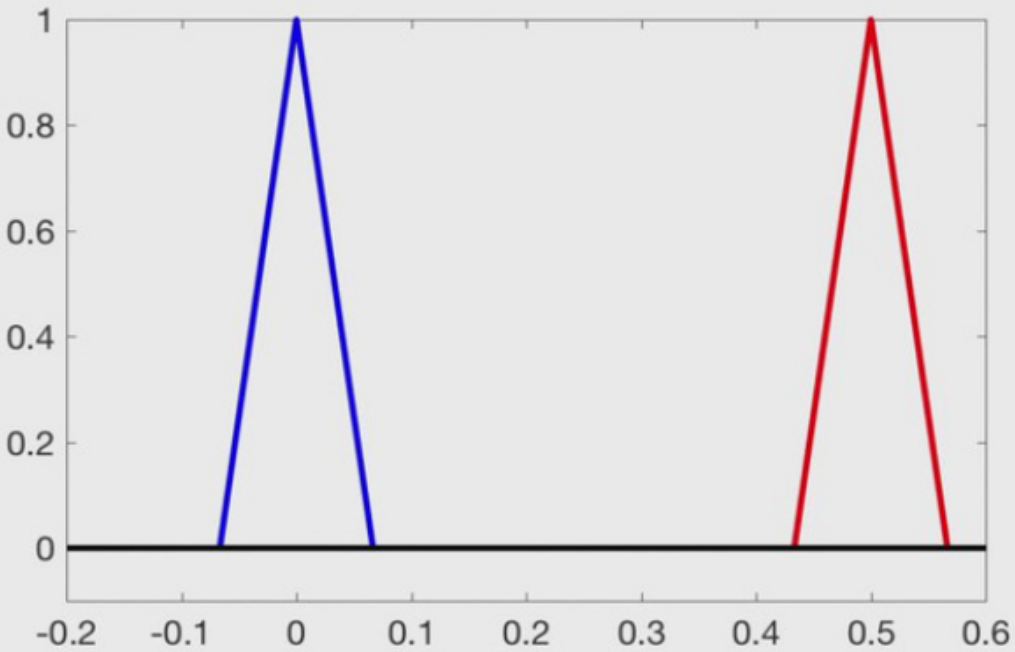
$$= \max_{T} (E_{x \sim p(x)}[T(x)] - E_{x \sim q(x)}[g(T(x))]) \tag{28}$$

How do ϕ-divergences behave?

Simple example: disjoint support, revisited.

Goodfellow et al. (NeurIPS 2014), Arjovsky and Bottou [ICLR 2017]

$$D_{KL}(Q, P) = \infty \qquad D_{JS}(P, Q) = \log 2$$



Intergal probability metric(IPM)

- Recall the f-divergence:

$$D_f(P||Q) = \max_T(E_{x\sim p(x)}[T(x)] - E_{x\sim q(x)}[g(T(x))]) \tag{29}$$

- Define the IPM as:

$$D_{IPM}(P||Q) = \sup_{f\in F}(E_{x\sim P}[f(x)] - E_{x\sim Q}[f(x)]) \tag{30}$$

- Why? Compare the 1-Moment:$E(x), E(y)$,2-Moment:$E(x^2), E[y^2]$.Hence you can understand that the IPM is the general case of them. When we choose different function f, the metric is different.

Maximum Mean Divergency

- It's a popular kinds of IMP. Recall the IMP:

$$D_{IPM}(P||Q) = \sup_{f \in F}(E_{x\sim P}[f(x)] - E_{x\sim Q}[f(x)]) \tag{31}$$

- MMD is defined as:

$$D_{IPM}(P||Q) = \sup_{f \in \text{unit ball in RKHS}}(E_{x\sim P}[f(x)] - E_{x\sim Q}[f(x)]) \tag{32}$$

- Why? Because this constrain help the MMD have a close-form solution.
- What is RKHS(Reproducing kernel Hilbert space)? I will quote some slides from Arthur Gretton(UCL Gatsby unit) to explain.

Hilbert space

Definition (Inner product)

Let $\mathcal{H}$ be a vector space over $\mathbb{R}$. A function $\langle \cdot, \cdot \rangle_{\mathcal{H}} : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ is an **inner product** on $\mathcal{H}$ if

1. Linear: $\langle \alpha_1 f_1 + \alpha_2 f_2, g \rangle_{\mathcal{H}} = \alpha_1 \langle f_1, g \rangle_{\mathcal{H}} + \alpha_2 \langle f_2, g \rangle_{\mathcal{H}}$
2. Symmetric: $\langle f, g \rangle_{\mathcal{H}} = \langle g, f \rangle_{\mathcal{H}}$
3. $\langle f, f \rangle_{\mathcal{H}} \geq 0$ and $\langle f, f \rangle_{\mathcal{H}} = 0$ if and only if $f = 0$.

Norm induced by the inner product: $\|f\|_{\mathcal{H}} := \sqrt{\langle f, f \rangle_{\mathcal{H}}}$

Definition (Hilbert space)

Inner product space containing Cauchy sequence limits.

Kernel

Definition

Let $\mathcal{X}$ be a non-empty set. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a **kernel** if there exists a Hilbert space $\mathcal{H}$ and a feature map $\phi : \mathcal{X} \to \mathcal{H}$ such that $\forall x, x' \in \mathcal{X}$,

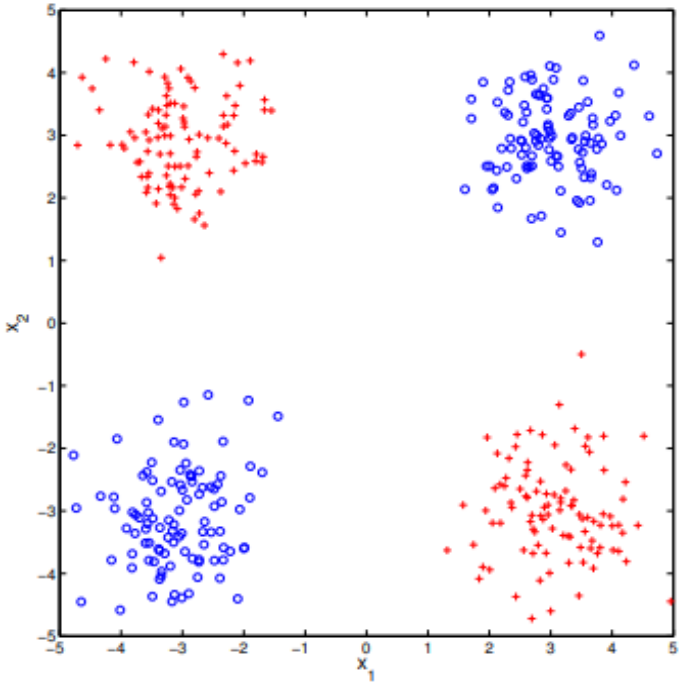
$$k(x, x') := \langle \phi(x), \phi(x') \rangle_{\mathcal{H}}.$$

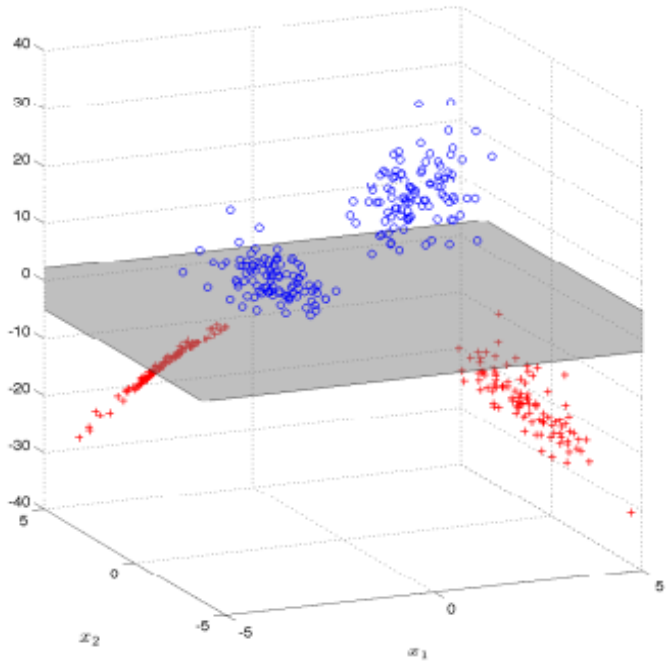
- Almost no conditions on $\mathcal{X}$ (eg, $\mathcal{X}$ itself doesn't need an inner product, eg. documents).
- A single kernel can correspond to several possible features. A trivial example for $\mathcal{X} := \mathbb{R}$:

$$\phi_1(x) = x \qquad \text{and} \qquad \phi_2(x) = \begin{bmatrix} x/\sqrt{2} \\ x/\sqrt{2} \end{bmatrix}$$

First example: finite space, polynomial features

Reminder: XOR example:





Example: finite space, polynomial features

Reminder: Feature space from XOR motivating example:

$$\phi : \mathbb{R}^2 \to \mathbb{R}^3$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \mapsto \phi(x) = \begin{bmatrix} x_1 \\ x_2 \\ x_1 x_2 \end{bmatrix},$$

with kernel

$$k(x, y) = \begin{bmatrix} x_1 \\ x_2 \\ x_1 x_2 \end{bmatrix}^\top \begin{bmatrix} y_1 \\ y_2 \\ y_1 y_2 \end{bmatrix}$$

(the standard inner product in $\mathbb{R}^3$ between features). Denote this feature space by $\mathcal{H}$.

Example: finite space, polynomial features

Define a **linear function** of the inputs x_1, x_2, and their product $x_1 x_2$,

$$f(x) = f_1 x_1 + f_2 x_2 + f_3 (x_1 x_2).$$

f in a space of functions mapping from $\mathcal{X} = \mathbb{R}^2$ to $\mathbb{R}$. Equivalent representation for f,

$$f(\cdot) = \begin{bmatrix} f_1 & f_2 & f_3 \end{bmatrix}^\top.$$

$f(\cdot)$ or f refers to the function as an object (here as a **vector** in $\mathbb{R}^3$)
$f(x) \in \mathbb{R}$ is function evaluated at a point (a **real number**).

$$f(x) = f(\cdot)^\top \phi(x) = \langle f(\cdot), \phi(x) \rangle_{\mathcal{H}}$$

Evaluation of f at x is an **inner product in feature space** (here standard inner product in $\mathbb{R}^3$)

$\mathcal{H}$ is a space of functions mapping $\mathbb{R}^2$ to $\mathbb{R}$.

Functions of infinitely many features

Functions are linear combinations of features:

$$f(x) = \langle f, \phi(x) \rangle_{\mathcal{H}} = \sum_{\ell=1}^{\infty} f_\ell \phi_\ell(x) = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \end{bmatrix}^\top \begin{bmatrix} \phi_1(x) \\ \phi_2(x) \\ \phi_3(x) \\ \vdots \end{bmatrix}$$

$$k(x, y) = \sum_{\ell=1}^{\infty} \phi_\ell(x) \phi_\ell(x')$$

$$f(x) = \sum_{\ell=1}^{\infty} f_\ell \phi_\ell(x) \qquad \sum_{\ell=1}^{\infty} f_\ell^2 < \infty.$$

The reproducing property

This example illustrates the two defining features of an RKHS:

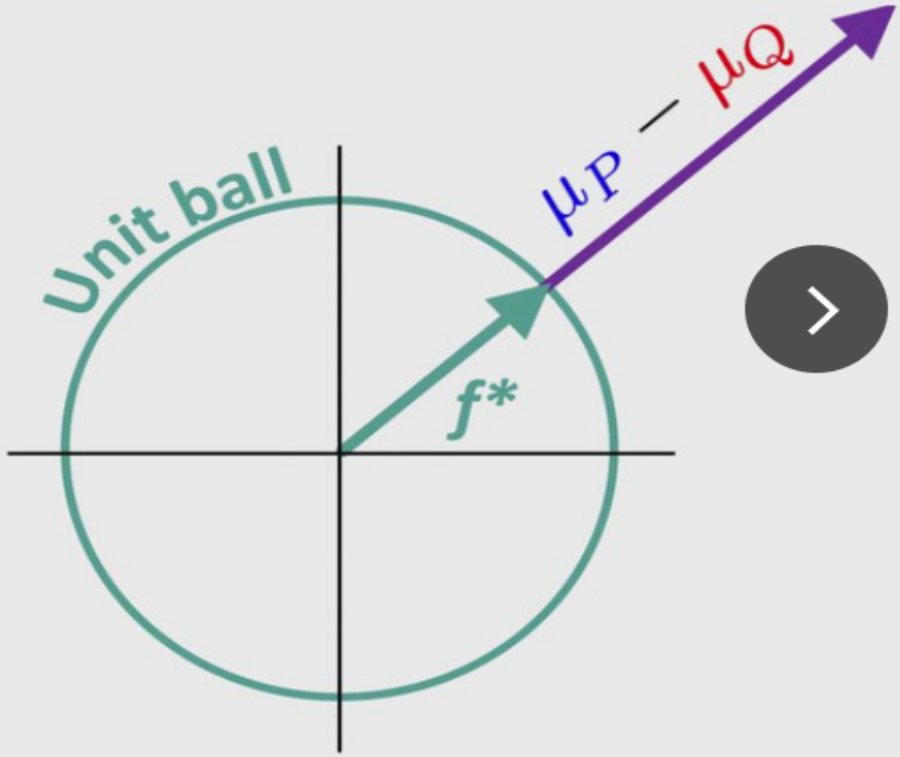
- **The reproducing property:** (kernel trick)
  $\forall x \in \mathcal{X}, \forall f(\cdot) \in \mathcal{H}, \quad \langle f(\cdot), k(\cdot, x)\rangle_{\mathcal{H}} = f(x)$
  ...or use shorter notation $\langle f, \phi(x)\rangle_{\mathcal{H}}$.
- The feature map of every point is a function: $k(\cdot, x) = \phi(x) \in \mathcal{H}$ for any $x \in \mathcal{X}$, and

$$k(x, x') = \left\langle \phi(x), \phi(x') \right\rangle_{\mathcal{H}} = \left\langle k(\cdot, x), k(\cdot, x') \right\rangle_{\mathcal{H}}.$$

Integral prob. metric vs feature mean difference

The MMD:

$$
\begin{aligned}
&MMD(P, Q; F) \\
&= \sup_{\|f\| \leq 1} \left[\mathbf{E}_P f(X) - \mathbf{E}_Q f(Y) \right] \\
&= \sup_{\|f\| \leq 1} \langle f, \mu_P - \mu_Q \rangle_{\mathcal{F}}
\end{aligned}
$$



$$f^* = \frac{\mu_P - \mu_Q}{\|\mu_P - \mu_Q\|}$$

Slide 28 / 276

17/34

Integral prob. metric vs feature mean difference

The MMD:

$$
\begin{aligned}
&MMD(P, Q; F) \\
&= \sup_{\|f\| \leq 1} [\mathbf{E}_P f(X) - \mathbf{E}_Q f(Y)] \\
&= \sup_{\|f\| \leq 1} \langle f, \mu_P - \mu_Q \rangle_{\mathcal{F}} \\
&= \|\mu_P - \mu_Q\|
\end{aligned}
$$

IPM view equivalent to feature mean difference (kernel case only)

$\mathcal{F}$ is an RKHS

- When $\mathcal{F} = \{f \in \mathcal{H} : \|f\|_{\mathcal{H}} \leq 1\}$, then

$$
\begin{aligned}
MMD^2(\mathbb{P}, \mathbb{Q}, \mathcal{F}) &= \left\| \overbrace{\int_X k(\cdot, x)\, d\mathbb{P}(x)}^{\mu_{\mathbb{P}}} - \overbrace{\int_X k(\cdot, x)\, d\mathbb{Q}(x)}^{\mu_{\mathbb{Q}}} \right\|_{\mathcal{H}}^2 \\
&= \overbrace{\int_X \int_X k(x, y)\, d\mathbb{P}(x)\, d\mathbb{P}(y)}^{\langle \mu_{\mathbb{P}}, \mu_{\mathbb{P}} \rangle_{\mathcal{H}}} \\
&\quad + \overbrace{\int_X \int_X k(x, y)\, d\mathbb{Q}(x)\, d\mathbb{Q}(y)}^{\langle \mu_{\mathbb{Q}}, \mu_{\mathbb{Q}} \rangle_{\mathcal{H}}} \\
&\quad - 2 \overbrace{\int_X \int_X k(x, y)\, d\mathbb{P}(x)\, d\mathbb{Q}(y)}^{\langle \mu_{\mathbb{P}}, \mu_{\mathbb{Q}} \rangle_{\mathcal{H}}} \\
&= \int_X \int_X k(x, y) d\mu(x)\, d\mu(y)
\end{aligned}
$$

for $\mu = \mathbb{P} - \mathbb{Q}$.

Not all Kernels are Useful

- $k(x, y) = c$ for all $x, y \in X$

$$MMD(\mathbb{P}, \mathbb{Q}, \mathcal{F}) = 0, \ \forall \mathbb{P}, \mathbb{Q}.$$

- Another example: $k(x, y) = \langle x, y \rangle_{\mathbb{R}^d}, \ x, y \in \mathbb{R}^d$

$$MMD(\mathbb{P}, \mathbb{Q}, \mathcal{F}) = \|M_{\mathbb{P}} - M_{\mathbb{Q}}\|_{\mathbb{R}^d},$$

where $M_{\mathbb{P}}$ is the mean of $\mathbb{P}$.

- Separable distributions can be made inseparable if the RKHS is not chosen properly.

How to choose $\mathcal{H}$?

Computation: RKHS vs. Other $\mathcal{F}$

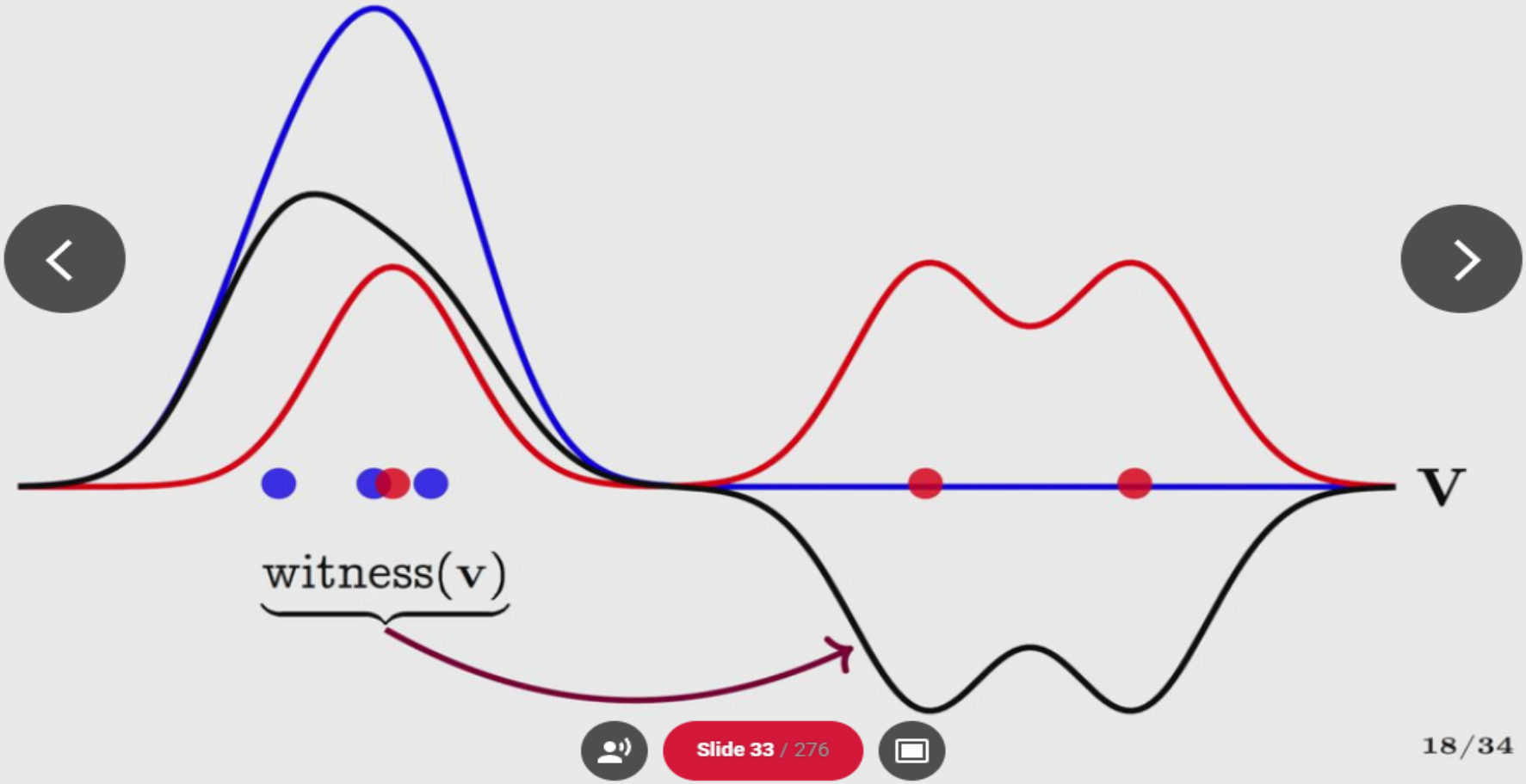
- Suppose $\{X_1, \ldots, X_m\} \overset{i.i.d.}{\sim} \mathbb{P}$ and $\{Y_1, \ldots, Y_n\} \overset{i.i.d.}{\sim} \mathbb{Q}$.
- Define $\mathbb{P}_m := \frac{1}{m}\sum_{i=1}^{m} \delta_{X_i}$ and $\mathbb{Q}_n := \frac{1}{n}\sum_{i=1}^{n} \delta_{Y_i}$, where δ_x represents the Dirac measure at x.
- $MMD(\mathbb{P}_m, \mathbb{Q}_n, \{\|f\|_{\mathcal{H}} \leq 1\})$ is obtained in a closed form as:

$$MMD^2(\mathbb{P}_m, \mathbb{Q}_n, \{\|f\|_{\mathcal{H}} \leq 1\}) = \frac{1}{m^2}\sum_{i,j=1}^{m} k(X_i, X_j) + \frac{1}{n^2}\sum_{i,j=1}^{n} k(Y_i, Y_j) - \frac{2}{mn}\sum_{i,j} k(X_i, Y_j).$$

Very easy to compute!!

Construction of MMD witness

Construction of empirical witness function (proof: next slide!)



Derivation of empirical witness function

Recall the witness function expression

$$f^* \propto \mu_P - \mu_Q$$

The empirical feature mean for P

$$\widehat{\mu}_P := \frac{1}{n}\sum_{i=1}^{n} \varphi(x_i)$$

The empirical witness function at v

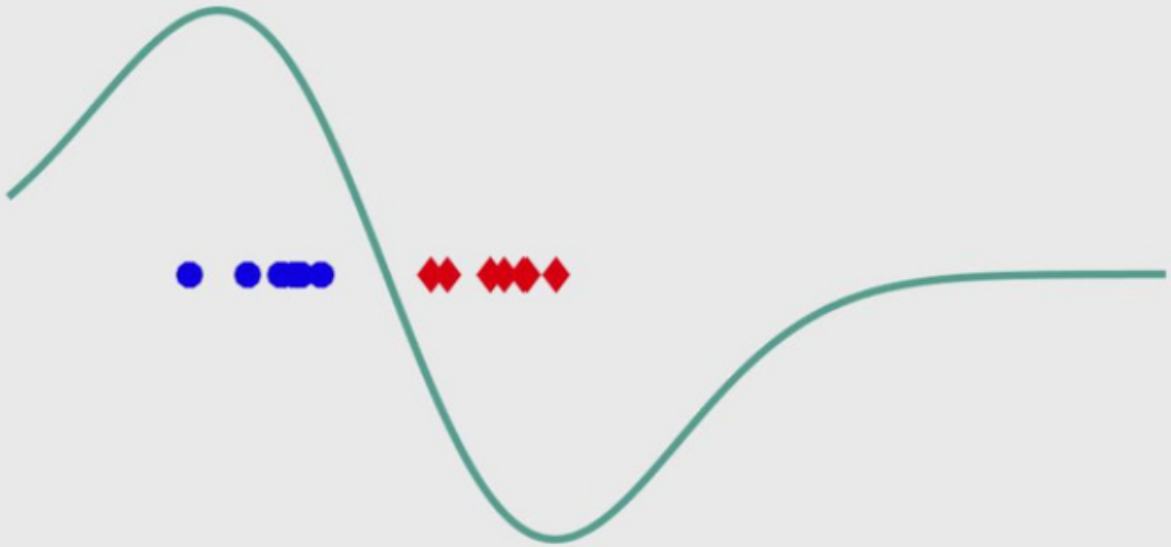
$$\begin{aligned} f^*(v) &= \langle f^*, \varphi(v) \rangle_{\mathcal{F}} \\ &\propto \langle \widehat{\mu}_P - \widehat{\mu}_Q, \varphi(v) \rangle_{\mathcal{F}} \\ &= \frac{1}{n}\sum_{i=1}^{n} k(x_i, v) - \frac{1}{n}\sum_{i=1}^{n} k(y_i, v) \end{aligned}$$

How does the MMD behave?

MMD with a broad kernel::

$$MMD(P, Q) = \sup_{\|f\|_{\mathcal{F}} \leq 1} E_P f(X) - E_Q f(Y)$$

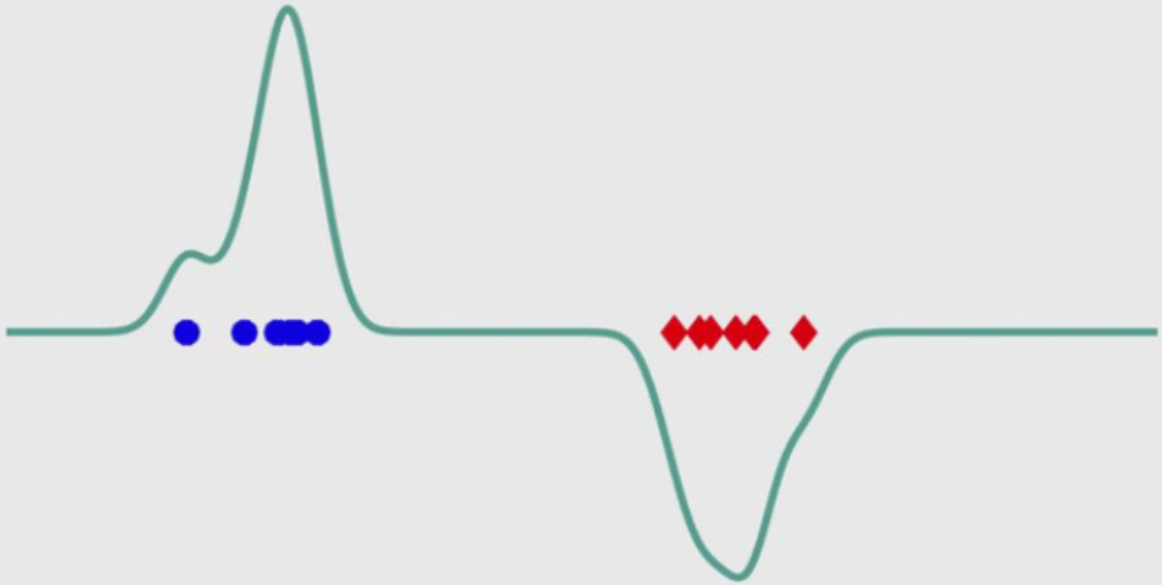
MMD=1.1



How does the MMD behave?

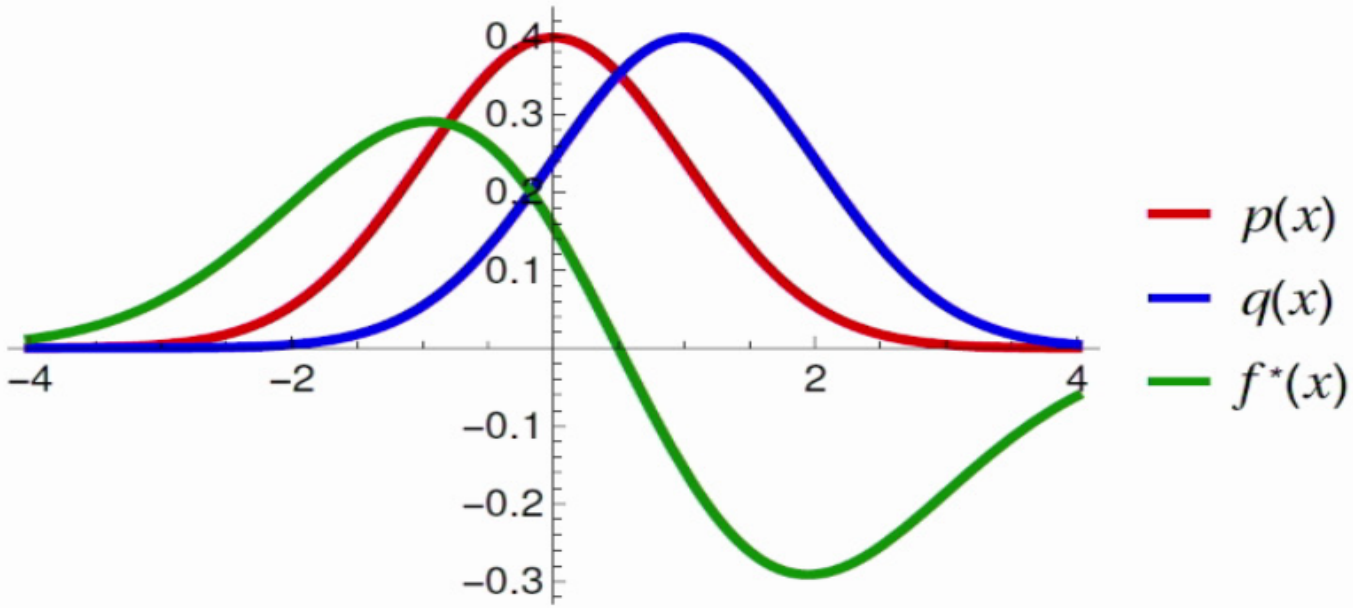
$MMD(P, Q)$ with a narrow kernel.

MMD=0.64



Statistical model criticism: toy example

$$\mathrm{MMD}(P, Q) = \sup_{\|f\|_{\mathcal{F}} \leq 1} [\mathbf{E}_q f - \mathbf{E}_p f]$$



Can we compute MMD with samples from Q and a **model** P?

Problem: usualy can't compute $\mathbf{E}_p f$ in closed form.

Slide 23 / 91

13/37

Stein idea

To get rid of $\mathbf{E}_p f$ in

$$\sup_{\|f\|_{\mathcal{F}} \leq 1} \left[\mathbf{E}_q f - \mathbf{E}_p f\right]$$

we define the (1-D) **Stein operator**

$$\left[\mathcal{A}_p f\right](x) = \frac{1}{p(x)} \frac{d}{dx}\left(f(x)p(x)\right)$$

Then

$$\mathbf{E}_p \mathcal{A}_p f = 0$$

subject to appropriate boundary conditions.

Gorham and Mackey (NeurIPS 15), Oates, Girolami, Chopin (JRSS B 2016)

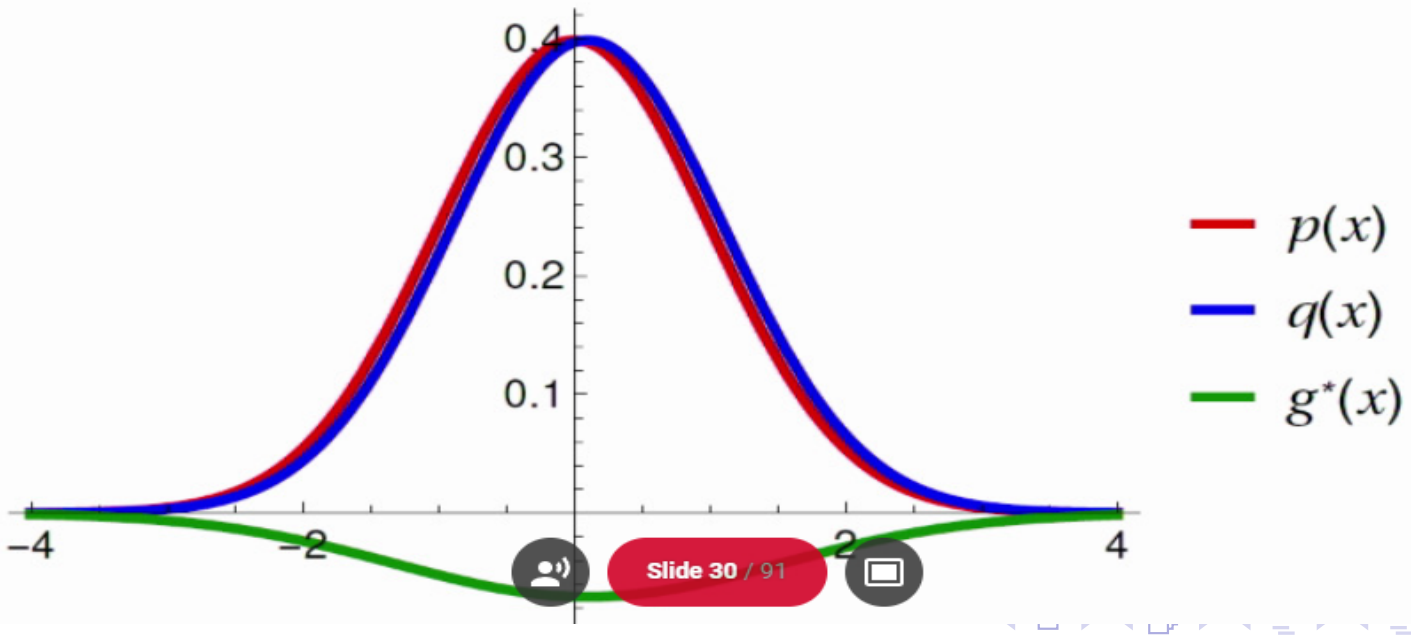
Kernel Stein Discrepancy

Stein operator

$$\mathcal{A}_p f = \frac{1}{p(x)} \frac{d}{dx} (f(x)p(x))$$

Kernel Stein Discrepancy (KSD)

$$\mathrm{KSD}_p(Q) = \sup_{\|g\|_{\mathcal{F}} \leq 1} \mathbf{E}_q \mathcal{A}_p g - \cancel{\mathbf{E}_p \mathcal{A}_p g} = \sup_{\|g\|_{\mathcal{F}} \leq 1} \mathbf{E}_q \mathcal{A}_p g$$



15/37

Simple expression using kernels

Re-write stein operator as:

$$[\mathcal{A}_p f](x) = \frac{1}{p(x)} \frac{d}{dx}(f(x)p(x))$$
$$= f(x)\frac{d}{dx}\log p(x) + \frac{d}{dx}f(x)$$

Can we define "Stein features"?

$$[\mathcal{A}_p f](x) = \left(\frac{d}{dx}\log p(x)\right) f(x) + \frac{d}{dx}f(x)$$
$$=: \langle f, \underbrace{\xi(x)}_{\text{stein features}} \rangle_{\mathcal{F}}$$

where $\mathbf{E}_{x\sim p}\xi(x) = 0$.

The kernel trick for derivatives

Reproducing property for the derivative: for differentiable $k(x, x')$,

$$\frac{d}{dx} f(x) = \left\langle f, \frac{d}{dx} \varphi(x) \right\rangle_{\mathcal{F}}$$

Using kernel derivative trick in (a),

$$\begin{aligned}
[\mathcal{A}_p f](x) &= \left(\frac{d}{dx} \log p(x) \right) f(x) + \frac{d}{dx} f(x) \\
&= \left\langle f, \left(\frac{d}{dx} \log p(x) \right) \varphi(x) + \underbrace{\frac{d}{dx} \varphi(x)}_{(a)} \right\rangle_{\mathcal{F}} \\
&=: \langle f, \xi(x) \rangle_{\mathcal{F}} .
\end{aligned}$$

Kernel stein discrepancy: derivation

Closed-form expression for KSD: given independent $x, x' \sim Q$, then

$$\begin{aligned}\mathrm{KSD}_p(Q) &= \sup_{\|g\|_{\mathcal{F}} \leq 1} \mathbf{E}_{x \sim q} \left([\mathcal{A}_p g](x) \right) \\ &= \sup_{\|g\|_{\mathcal{F}} \leq 1} \mathbf{E}_{x \sim q} \langle g, \xi_x \rangle_{\mathcal{F}} \\ &\underset{(a)}{=} \sup_{\|g\|_{\mathcal{F}} \leq 1} \langle g, \mathbf{E}_{x \sim q} \xi_x \rangle_{\mathcal{F}} = \|\mathbf{E}_{x \sim q} \xi_x\|_{\mathcal{F}}\end{aligned}$$

Caution: (a) requires a condition for the Riesz theorem to hold,

$$\mathbf{E}_{x \sim q} \left(\frac{d}{dx} \log p(x) \right)^2 < \infty.$$

Chwialkowski, Strathmann, G., (ICML 2016) Liu, Lee, Jordan (ICML 2016)

What happen up to now?

We already introduce two type of distance:

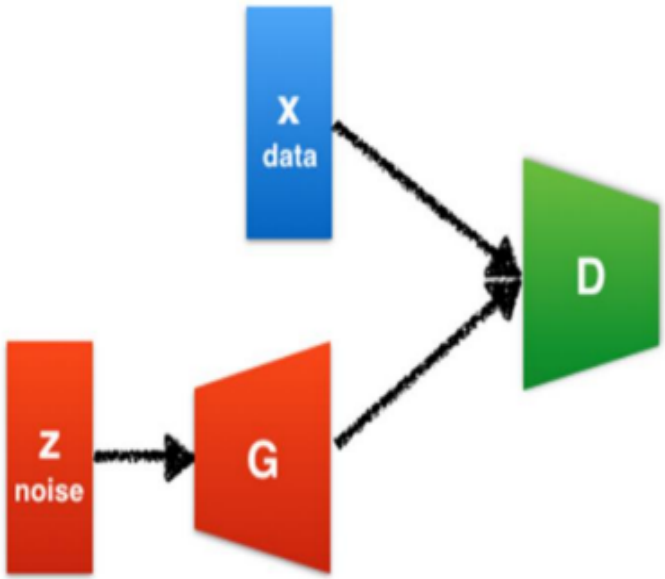
- f-divergence:KL,JS
- IPM: MMD , KSD
- Why we need to measure the distance between data and data(two-sample test), or between data and model(goodness of fit)?

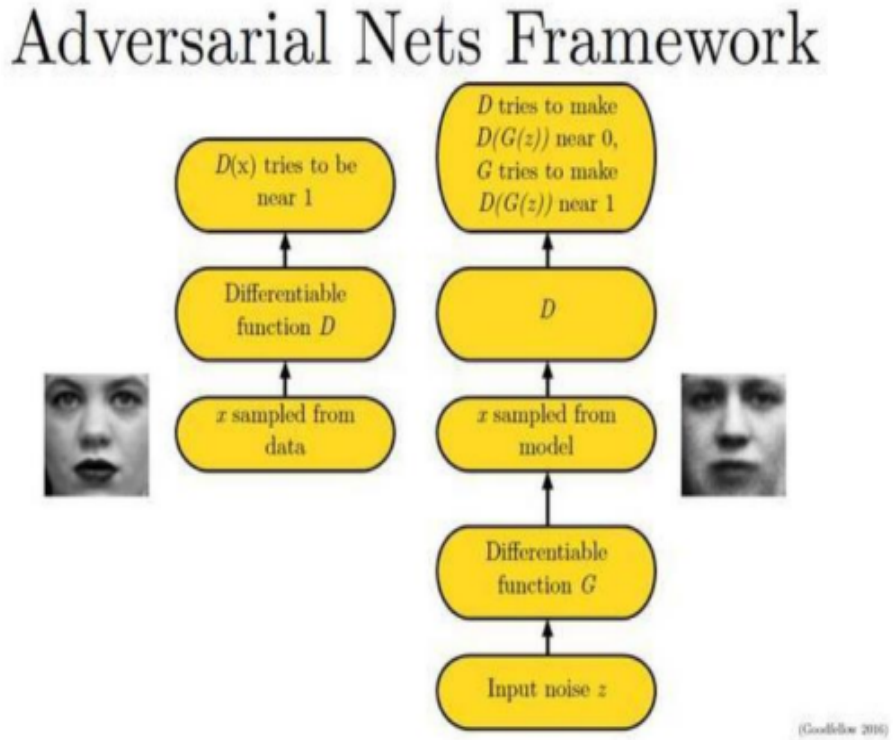
Generative Adversarial Nets

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio
Département d'informatique et de recherche opérationnelle
Université de Montréal
Montréal, QC H3C 3J7

GANs（Ian J.Goodfellow 2014）

(对抗生成网络)





GANs (Ian J.Goodfellow 2014)

(对抗生成网络)

Generative Adversarial Nets

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio
Département d'informatique et de recherche opérationnelle
Université de Montréal
Montréal, QC H3C 3J7

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)\right].$$

end for

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Update the generator by descending its stochastic gradient:

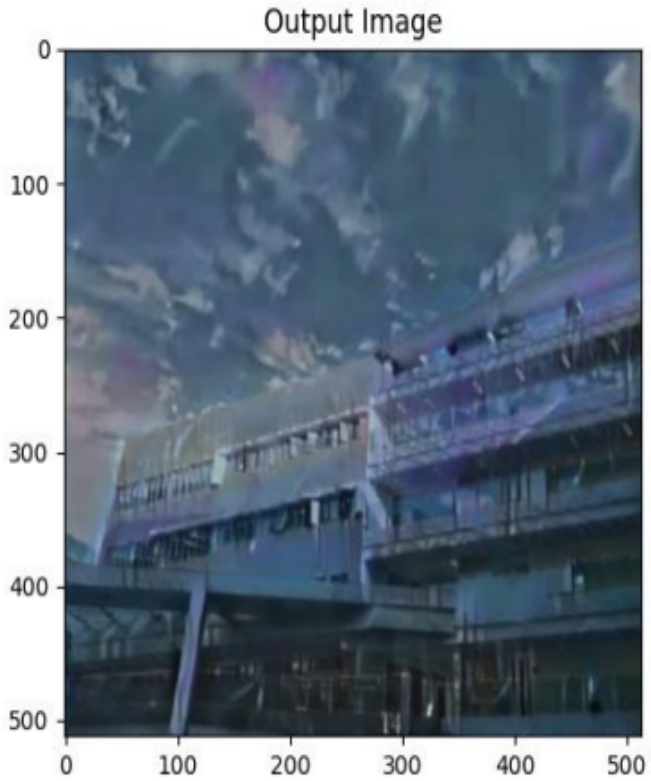
$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

$$\nabla C \approx \frac{1}{m} \sum_{j=1}^{m} \nabla C_{X_j},$$

Implementation of Gaty's Model (Base on Pytorch)



Part5. Future work

Representativeness in Machine Learning

A GOODNESS OF FIT MEASURE FOR GENERATIVE NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

We define a goodness of fit measure for generative networks which captures how well the network can generate the training data, which is necessary to learn the true data distribution. We demonstrate how our measure can be leveraged to understand mode collapse in generative adversarial networks and provide practitioners with a novel way to perform model comparison and early stopping without having to access another trained model as with Frechet Inception Distance or Inception Score. This measure shows that several successful, popular generative models, such as DCGAN and WGAN, fall very short of learning the data distribution. We identify this issue in generative models and empirically show that overparameterization via subsampling data and using a mixture of models improves performance in terms of goodness of fit.

Representativeness in Machine Learning

Reforming Generative Autoencoders via Goodness-of-Fit Hypothesis Testing

Aaron Palmer
Computer Science Dept.
University of Connecticut
Storrs, CT 06269

Dipak K. Dey
Statistics Dept.
University of Connecticut
Storrs, CT 06269

Jinbo Bi
Computer Science Dept.
University of Connecticut
Storrs, CT 06269

Abstract

Generative models, while not new, have taken the deep learning field by storm. However, the widely used training methods have not exploited the substantial statistical literature concerning parametric distributional testing. Having sound theoretical foundations, these goodness-of-fit tests enable parts of the black box to be stripped away. In this paper we use the Shapiro-Wilk and propose a new multivariate generalization of Shapiro-Wilk to respectively test for univariate and multivariate normality of the code layer of a generative autoencoder. By replacing the discriminator in traditional deep models with the hypothesis tests, we gain several advantages: objectively evaluate whether the encoder is actually embedding data onto a normal manifold, accurately define when convergence happens, explicitly balance between reconstruction and encoding training. Not only does our method produce competitive results, but it does so in a fraction of the time. We highlight the fact that the hypothesis tests used in our model asymptotically lead to the same solution of the L_2-Wasserstein distance metrics used by several generative models today.

concerning parametric distributional hypothesis testing with a solid theoretical base. One particular group of deep generative models which, we show in this study, can benefit from hypothesis testing is the generative autoencoder (GAE). The objective of these models is to reconstruct the input as accurately as possible, while constraining the code layer to a specified distribution, usually normal. Often times once training has ended, this code layer distribution does not in fact match the required distribution. The spirit of these models is to embed data into a distribution that matches the prior to enable sampling, and thus it is of utmost importance we have ways to assess the quality of the fit. In other words, if the embedded distribution does not match the prior that is used to sample and generate instances, the method does not work in theory.

In this paper we propose to use goodness-of-fit hypothesis tests of normality on the code layer of an autoencoder as a new type of critic in both the univariate and multivariate case. Doing so leads to an adversary-free optimization problem. These hypothesis tests provide a more direct way to measure if the data representation, the latent code layer, matches a pre-specified distribution. More specifically, we test for normality using a composite test:

$$H_0 : \mathbf{X} \in \mathcal{G} \qquad \text{vs} \qquad H_1 : \mathbf{X} \notin \mathcal{G} \tag{1}$$

where $\mathcal{G} = \{\pi : \pi = \mathcal{N}(\mu, \mathbf{\Sigma}), -\infty < \mu < \infty, \mathbf{\Sigma}$ is positive semi-definite (p.s.d)$\}$. Many tests for comparing two distributions can be used in our model[1]. We specifically focus on the well studied univariate