

```

In[*]:= (*导入数据*)
Clear[data, i, L, n, temp];
清除
L = 100; n = 25; data = Table[0, {i, L}];
表格

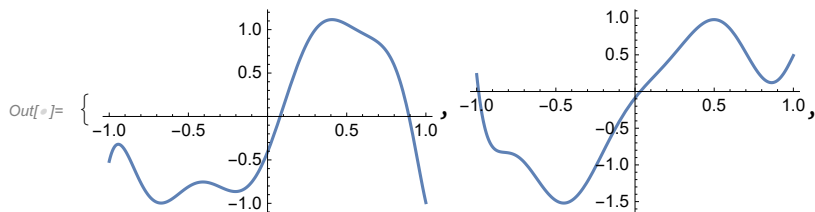
For[i = 1, i ≤ L, i++,
For循环
temp = StringJoin[
连接字符串
"E:\\study_materials\\MachineLearning\\HW2\\data\\data_", ToString[i]];
自然常数 转换为字符串
data[[i]] = Import[temp, "Table"];
导入 表格
];

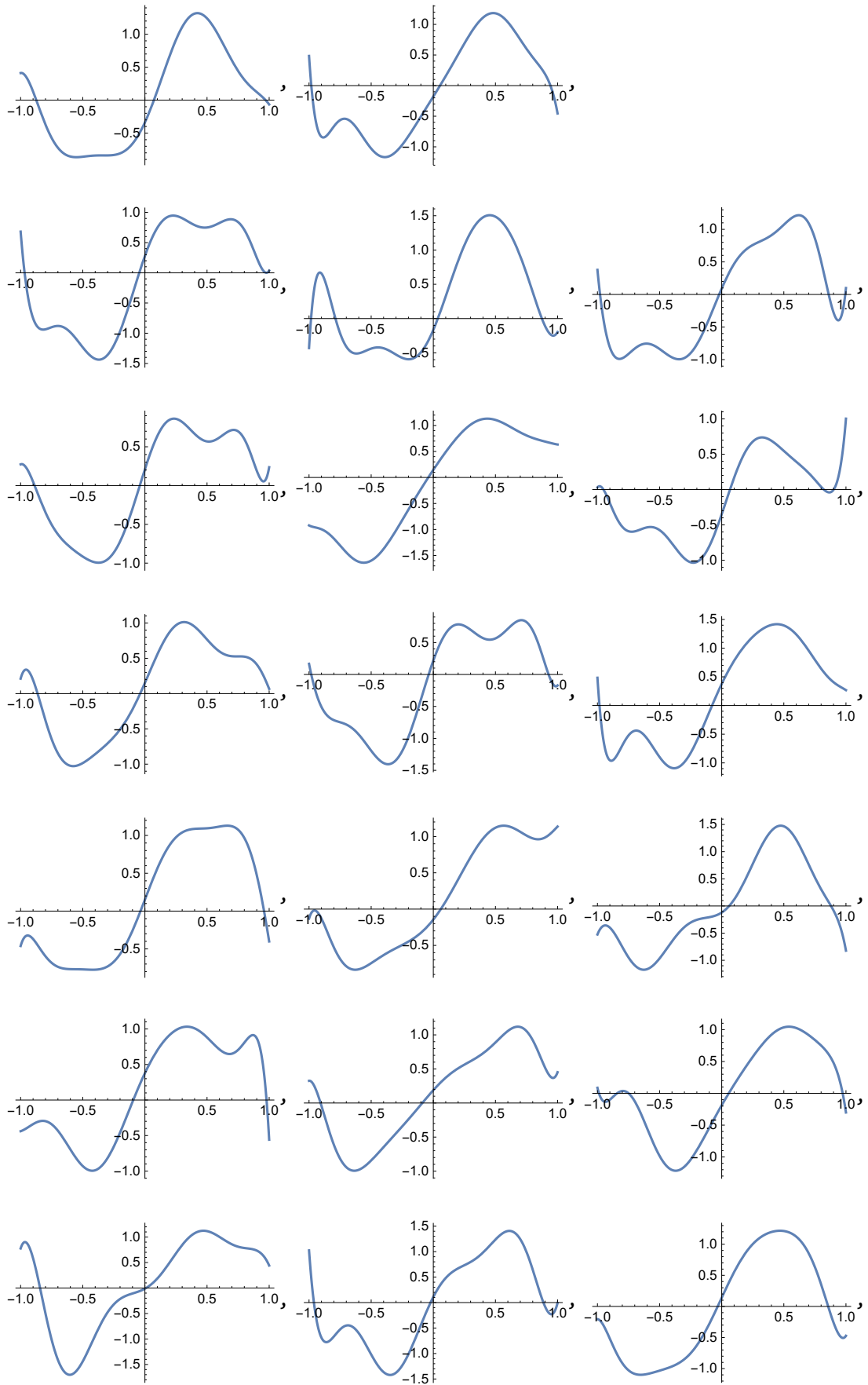
(*2.lambda=10^(-10)*)
phi[i_, x_] := N[E^(-(x - 0.2 (i - 12.5))^2), 20]
自然常数

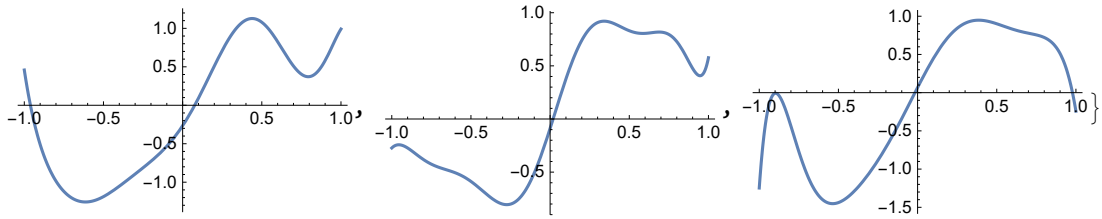
In[*]:= lambda = 10^(-10);
img = Table[0, {i, L}];
表格

For[l = 1, l ≤ L, l++,
For循环
tempx = data[[l, All, 1]];
全部
tempy = data[[l, All, 2]];
全部
tempPhi = Table[phi[i, tempx], {i, 24}];
表格
tempPhi = Transpose[PrependTo[tempPhi, Table[1, {i, 25}]]];
转置 前加于 表格
tempw = Inverse[(lambda * IdentityMatrix[25] + Transpose[tempPhi].tempPhi)].
逆 单位矩阵 转置
Transpose[tempPhi].tempy;
转置
img[[l]] = Plot[Sum[tempw[[i + 1]] * phi[i, x], {i, 24}] + tempw[[1]], {x, -1, 1}]
绘图 求和
];
Table[img[[i]], {i, 25}]
表格

```





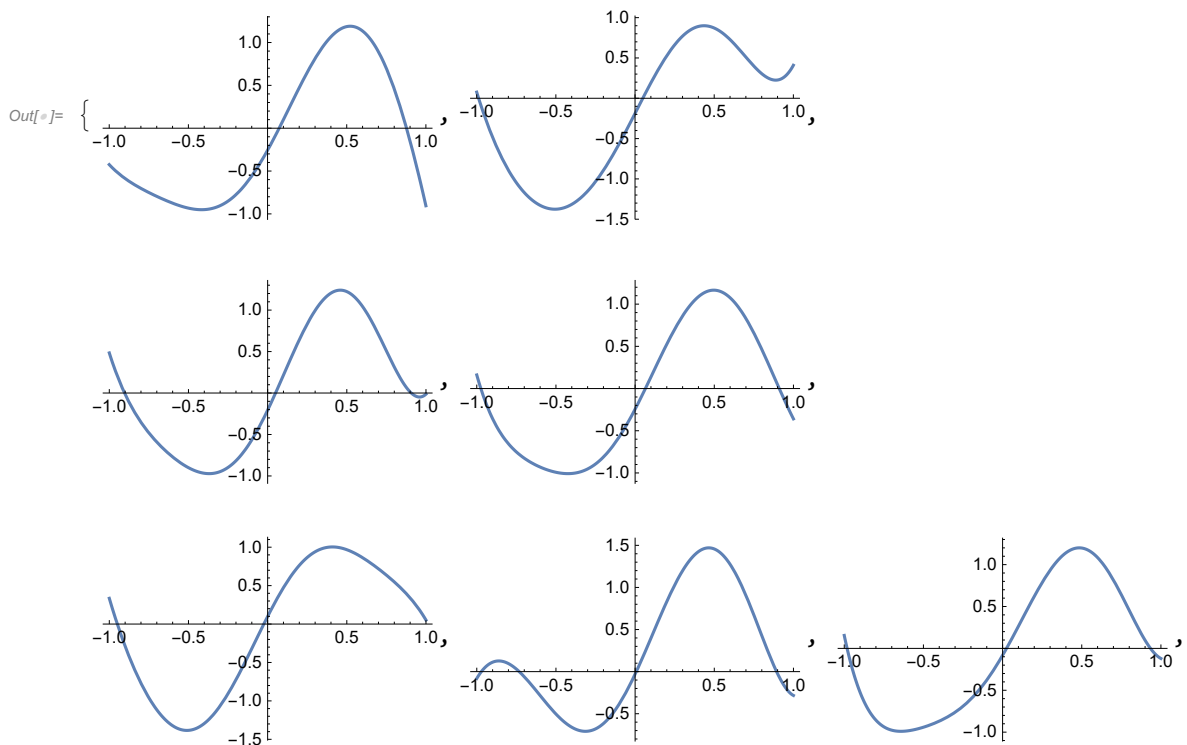


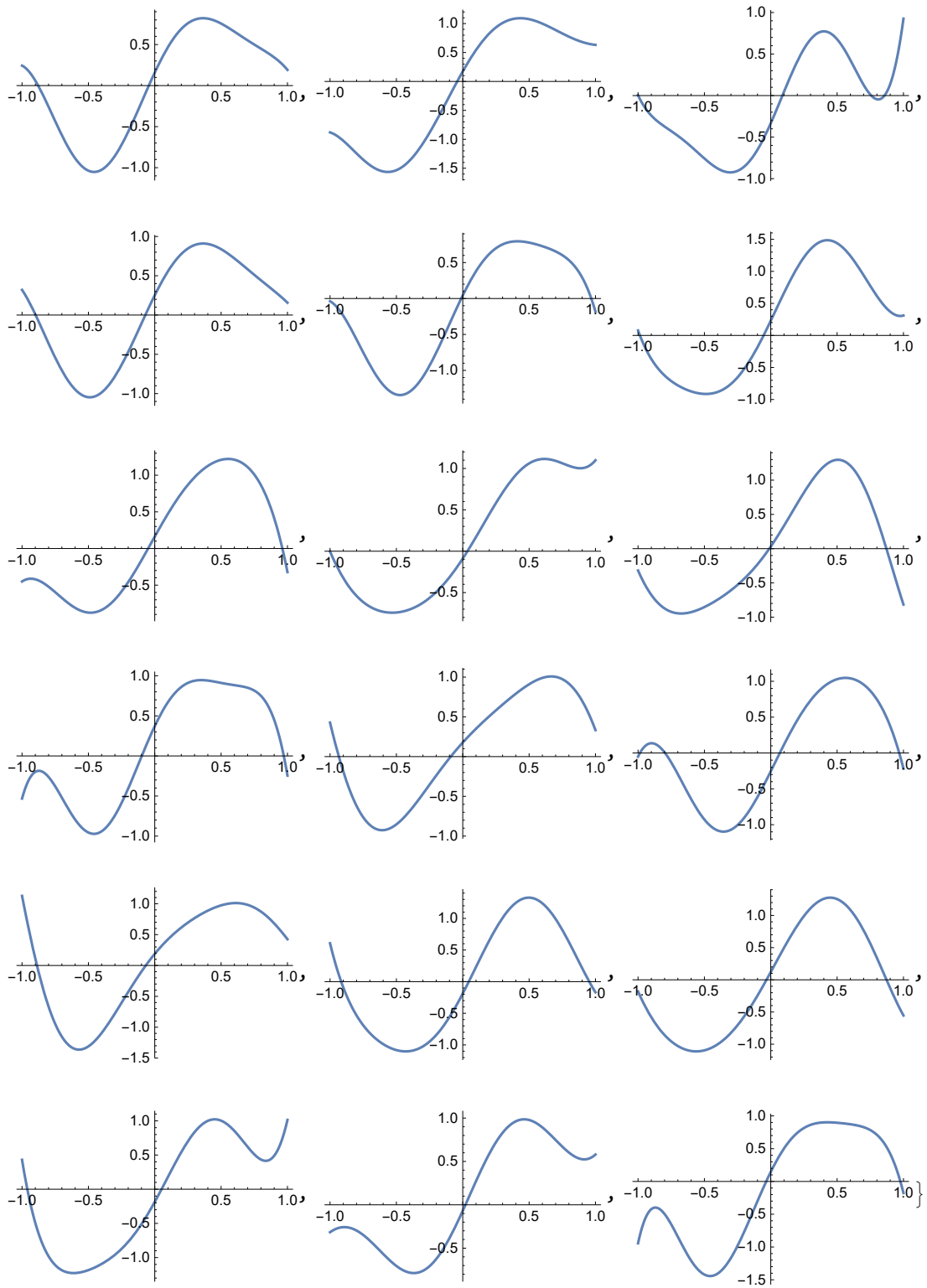
(\*2.lambda=10^(-5)\*)

```

In[ ]:= lambda = 10^(-5);
img = Table[0, {i, L}];
      | 表格
For[1 = 1, 1 ≤ L, 1++,
  | For循环
    tempx = data[[1, All, 1]];
      | 全部
    tempy = data[[1, All, 2]];
      | 全部
    tempPhi = Table[phi[i, tempx], {i, 24}];
      | 表格
    tempPhi = Transpose[PrependTo[tempPhi, Table[1, {i, 25}]]];
      | 转置 | 前加于 | 表格
    tempw = Inverse[(lambda * IdentityMatrix[25] + Transpose[tempPhi].tempPhi)];
      | 逆 | 单位矩阵 | 转置
    Transpose[tempPhi].tempy;
      | 转置
    img[[1]] = Plot[Sum[tempw[[i + 1]] * phi[i, x], {i, 24}] + tempw[[1]], {x, -1, 1}]
      | 绘图 | 求和
  ];
Table[img[[i]], {i, 25}]
      | 表格

```



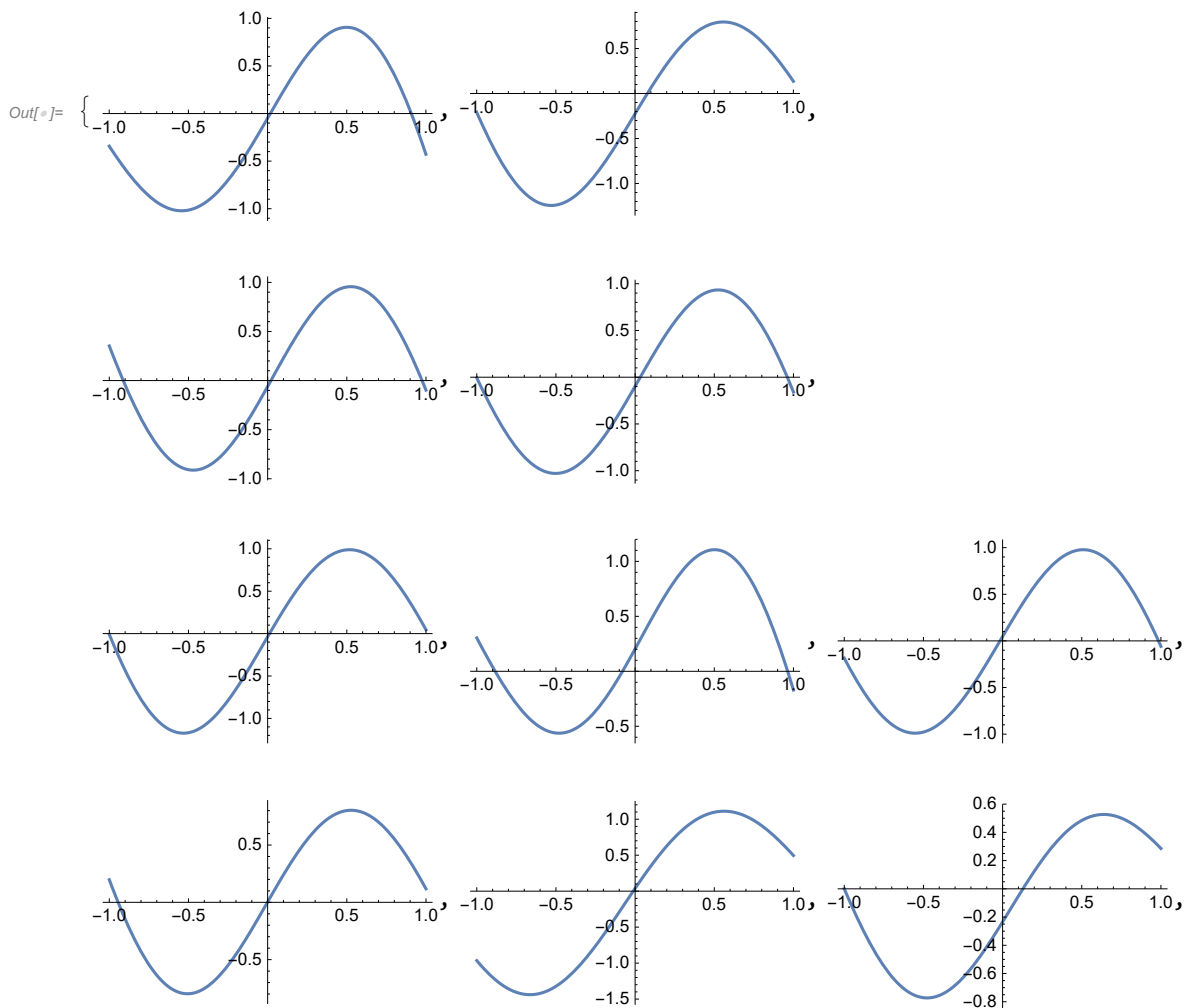


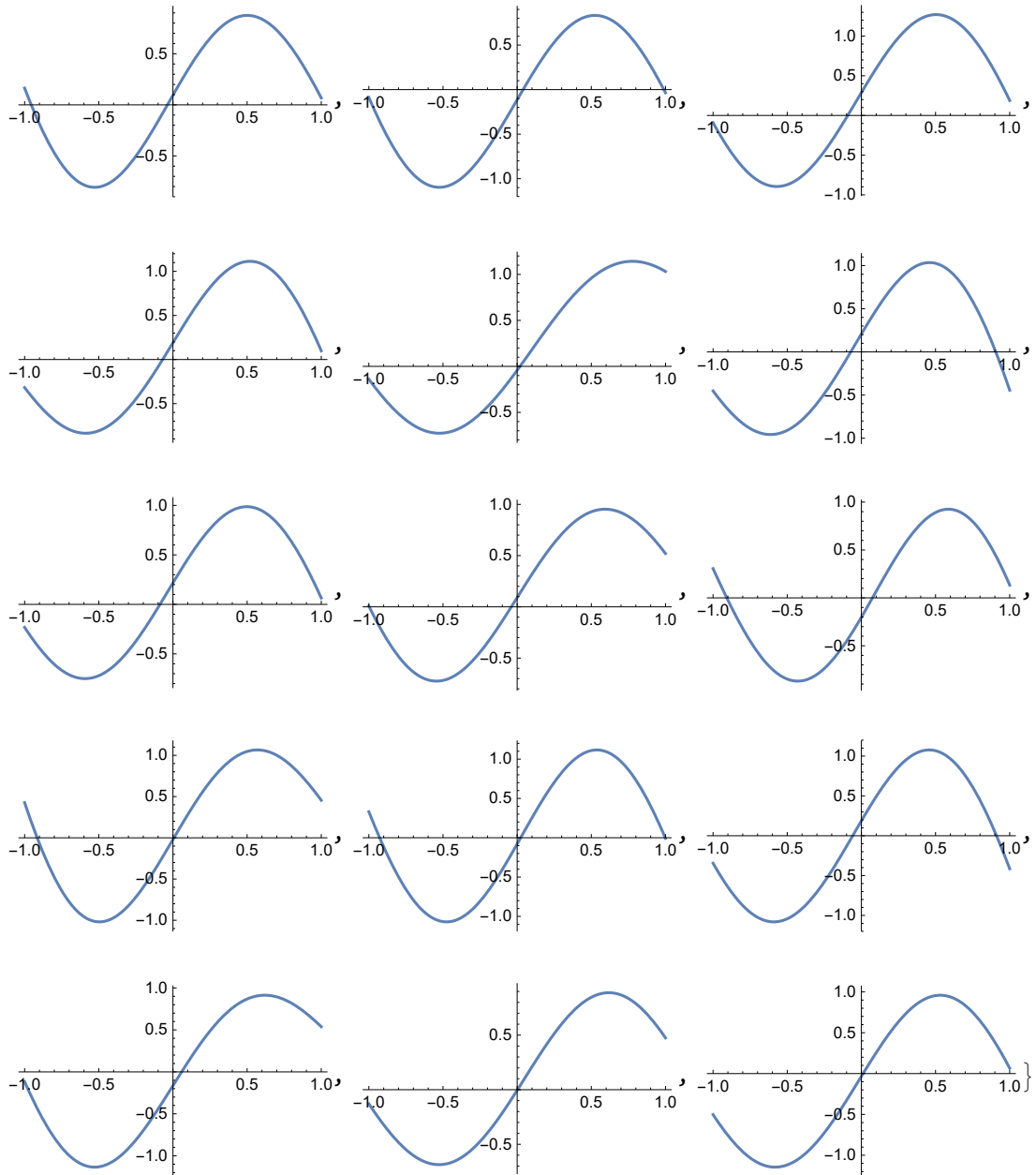
(\*2.lambd=10<sup>-1</sup>\*)

```

In[ ]:= lambda = 10^(-1);
img = Table[0, {i, L}];
      | 表格
For[1 = 1, 1 ≤ L, 1++,
  | 循环
    tempx = data[[1, All, 1]];
              | 全部
    tempy = data[[1, All, 2]];
              | 全部
    tempPhi = Table[phi[i, tempx], {i, 24}];
              | 表格
    tempPhi = Transpose[PrependTo[tempPhi, Table[1, {i, 25}]]];
              | 转置      | 前加于      | 表格
    tempw = Inverse[(lambda * IdentityMatrix[25] + Transpose[tempPhi].tempPhi)].
              | 逆      | 单位矩阵      | 转置
              Transpose[tempPhi].tempy;
              | 转置
    img[[1]] = Plot[Sum[tempw[[i + 1]] * phi[i, x], {i, 24}] + tempw[[1]], {x, -1, 1}]
              | 绘图 | 求和
];
Table[img[[i]], {i, 25}]
      | 表格

```



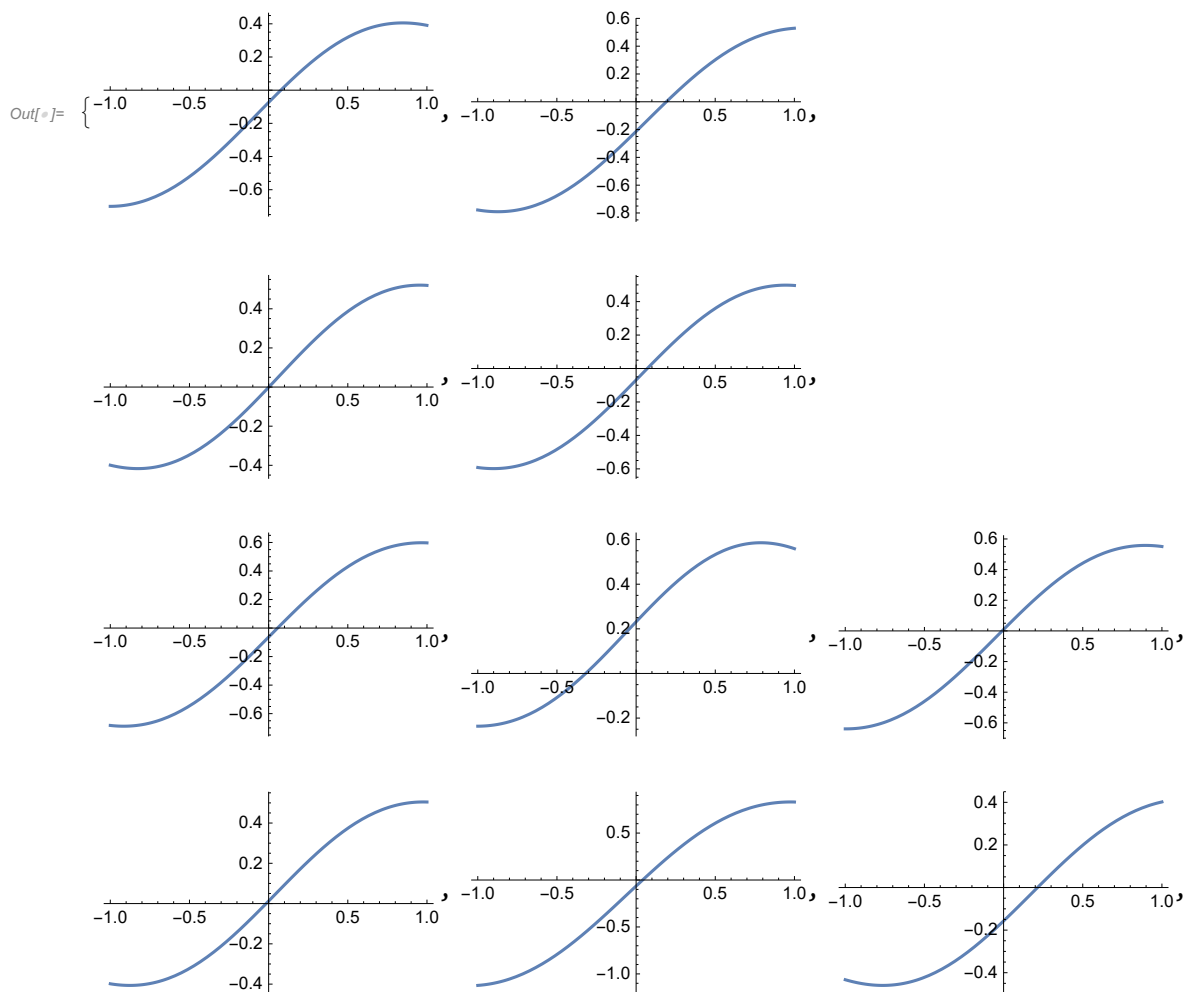


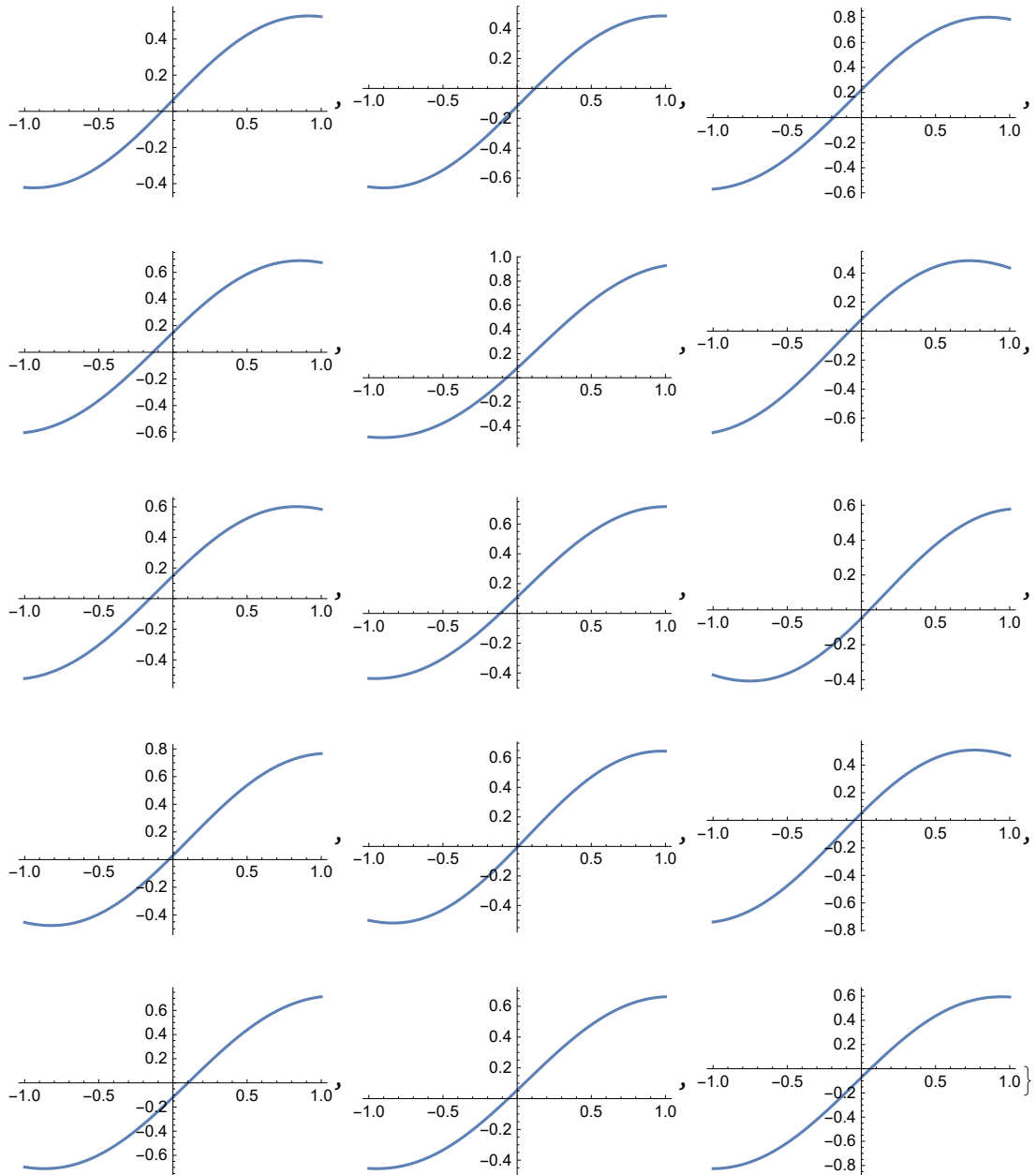
( $\lambda=10$ )

```

In[ ]:= lambda = 10^(1);
img = Table[0, {i, L}];
      | 表格
For[1 = 1, 1 ≤ L, 1++,
  | For循环
  tempx = data[[1, All, 1]];
      | 全部
  tempy = data[[1, All, 2]];
      | 全部
  tempPhi = Table[phi[i, tempx], {i, 24}];
      | 表格
  tempPhi = Transpose[PrependTo[tempPhi, Table[1, {i, 25}]]];
      | 转置      | 前加于      | 表格
  tempw = Inverse[(lambda * IdentityMatrix[25] + Transpose[tempPhi].tempPhi)];
      | 逆      | 单位矩阵      | 转置
  Transpose[tempPhi].tempy;
      | 转置
  img[[1]] = Plot[Sum[tempw[[i + 1]] * phi[i, x], {i, 24}] + tempw[[1]], {x, -1, 1}]
      | 绘图 | 求和
];
Table[img[[i]], {i, 25}]
      | 表格

```





```

In[ ]:= (*3.*)
Clear[h, lambda, tempx, tempy, tempPhi, tempw, img, l, i];
清除
h[x_] := Sin[Pi x];
正弦 圆周率
xn = data[[1, All, 1]];
全部

```



```

In[ ]:= m = 400;
deltax = N[4 / m];
lam = Table[-3 + i deltax, {i, 0, m}];
list = Table[{lam[[i]], 0, 0, 0}, {i, m + 1}];
For[k = 1, k ≤ m + 1, k++,
  yx = 0;
  lambda = 10^ (lam[[k]]);
  tempyx = 0;
  For[l = 1, l ≤ L, l++,
    tempx = data[[l, All, 1]];
    tempy = data[[l, All, 2]];
    tempPhi = Table[phi[i, tempx], {i, 24}];
    tempPhi = Transpose[PrependTo[tempPhi, Table[1, {i, 25}]]];
    tempw = Inverse[(lambda * IdentityMatrix[25] + Transpose[tempPhi].tempPhi)].
    Transpose[tempPhi].tempy;
    tempyx += (Sum[tempw[[i + 1]] * phi[i, x], {i, 24}] + tempw[[1]]);
  ];
  (*生成平均估计*)
  tempyx = tempyx / L;
  (*生成bias^2*)
  biasdiff = (tempyx /. x → xn) - h[xn];
  list[[k, 2]] = Sum[(biasdiff[[i]])^2, {i, 1, n}] / n;
  (*生成variance*)
  list[[k, 3]] =
    Sum[(data[[l, i, 2]] - tempyx /. x → xn[[i]])^2, {l, 100}, {i, 25}] / n / L;
  (*生成bias^2+variance*)
  list[[k, 4]] = list[[k, 2]] + list[[k, 3]];
]

```

In[ ]:= (\*绘图\*)

ListLinePlot[{list[[All, {1, 2}]], list[[All, {1, 3}]], list[[All, {1, 4}]]},

绘制点集的线条

全部

全部

全部

PlotLegends → {"bias^2", "variance", "bias^2+variance"}]

绘图的图例

