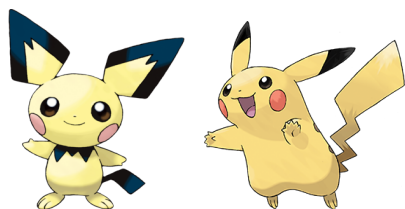


## Labb 2

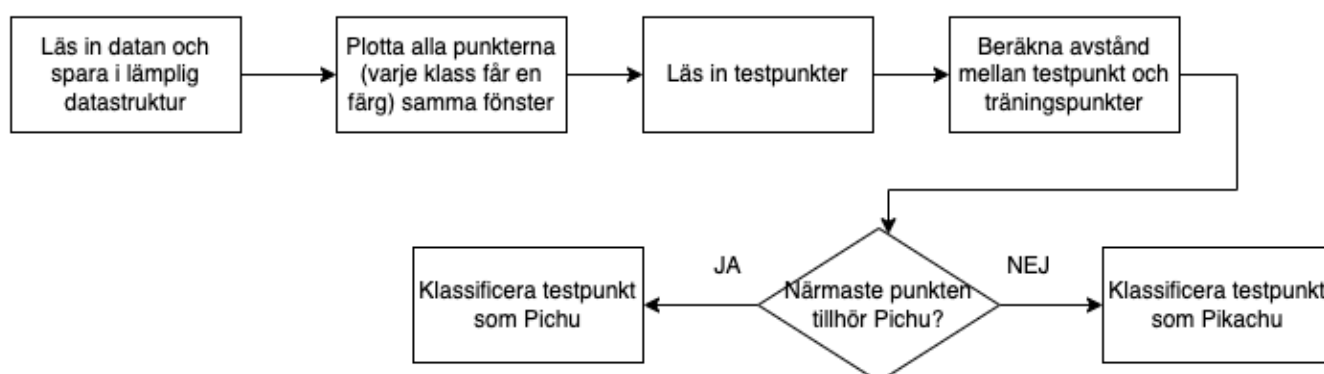
**Syftet** med den här laborationen är att använda verktygen du lärt dig i Python för att implementera en förenklad maskininlärningsalgoritm.



I den här laborationen finns (simulerad) data på Pichus och Pikachus längder och bredder. Du ska skapa en algoritm som baserat på den givna datan kunna avgöra om en ny datapunkt ska klassificeras som Pichu eller Pikachu.

### Grunduppgift

Följ detta flödesschema för att bygga den grundläggande algoritmen



(25,32), (24.2,31.5), (22,34), (20.5,34)

Facit på givna testdatan:

```
Sample with (width, height): (25, 32) classified as Pikachu
Sample with (width, height): (24.2, 31.5) classified as Pikachu
Sample with (width, height): (22, 34) classified as Pikachu
Sample with (width, height): (20.5, 34) classified as Pichu
```

### Uppgifter

Gör dessa uppgifter när du klassificerat testdatan korrekt.

1. Låt användaren mata in en testpunkt och låt algoritmen avgöra dess klass. Ta med felhanteringen som tar hand om negativa tal och icke-numeriska inputs. Se till att ha användarvänliga

felmeddelanden.

- Den approachen vi använt med närmaste punkten kan klassificera fel när punkterna för respektive klass går in i varandra. Nu ska du istället välja de tio närmaste punkterna till din testpunkt. Den klass testpunkten tillhör avgörs av majoritetsröstning av de närmaste punkterna.

---

## Bonusuppgifter (frivilliga)

- Dela in ursprungsdatan slumpmässigt så att:

- 100 är träningsdata (50 Pikachu, 50 Pichu)
- 50 är testdata (25 Pikachu, 25 Pichu)

- Beräkna noggrannheten genom följande formel:

$$\text{accuracy} = (\#TP + \#TN) / (\text{total})$$

där

	Pikachu actual	Pichu actual
Pikachu predicted	TP	FP
Pichu predicted	FN	TN

Här har vi låtit Pikachu vara positiv och Pichu vara "icke-Pikachu" dvs negativ.

Notera: För varje testpunkt beräknar vi distansen till träningspunkterna

- Upprepa uppgift 3 och 4 tio gånger, plotta en graf över accuracy och rapportera medelaccuracy.

---

## Bedömning

Om du har fått någon kodsnuitt från någon annan eller hittat i någon sida är det **viktigt** att du källhänvisar. Skriv en kommentar bredvid koden som du har tagit.

### Godkänt

- löst grunduppgift, samt övriga uppgifter på korrekt sätt
- koden är kommenterad med relevanta kommentarer
- variabelnamnen är bra valda
- gjort flera relevanta git commits

### Väl Godkänt

- koden är enkel att följa
- koden är effektiv

- koden är välstrukturerad med lämpliga funktioner
- kommentarerna är datavetenskapligt korrekta
- löst samtliga uppgifter