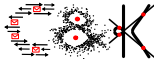
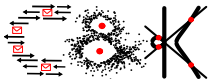


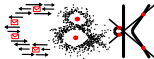
# Lossless compression



- $A = \{a_1, \dots, a_d\}$  finite alphabet,  $p = (p_1, \dots, p_d) \in S^d$ , i.e. probability distribution
- $X = X_1 \cdots X_l \in A^*$
- $\forall j, i : \Pr[X_j = a_i] = p_i$



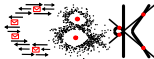
# Lossless compression



- $A = \{a_1, \dots, a_d\}$  finite alphabet,  $p = (p_1, \dots, p_d) \in S^d$ , i.e. probability distribution
- $X = X_1 \cdots X_l \in A^*$
- $\forall j, i : \Pr[X_j = a_i] = p_i$

**Want** function  $f : A \rightarrow \{0, 1\}^*$  such that

- 1  $\forall i, j, i \neq j : f(a_i)$  is not a prefix of  $f(a_j)$
- 2  $E[f] := \sum p_i |f(a_i)|$  is small

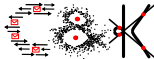


- $A = \{a_1, \dots, a_d\}$  finite alphabet,  $p = (p_1, \dots, p_d) \in S^d$ , i.e. probability distribution
- $X = X_1 \cdots X_l \in A^*$
- $\forall j, i : \Pr[X_j = a_i] = p_i$

**Want** function  $f : A \rightarrow \{0, 1\}^*$  such that

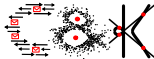
- 1  $\forall i, j, i \neq j : f(a_i)$  is not a prefix of  $f(a_j)$
  - 2  $E[f] := \sum p_i |f(a_i)|$  is small
- 
- 1 guarantees that  $X$  can be recovered from  $f(X) = f(X_1) \cdots f(X_l)$
  - 2  $E[f]$  called expected codeword length of  $f$

# Shannon code



Given  $A = \{a_1, \dots, a_d\}$ , Shannon code  $S : A \rightarrow \{0, 1\}^*$  achieves

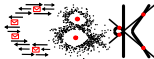
- 1  $\forall i : |S(a_i)| = \lceil \log(1/p_i) \rceil$
- 2  $E[S] = \sum p_i \lceil \log(1/p_i) \rceil$



Given  $A = \{a_1, \dots, a_d\}$ , Shannon code  $S : A \rightarrow \{0, 1\}^*$  achieves

- 1  $\forall i : |S(a_i)| = \lceil \log(1/p_i) \rceil$
- 2  $E[S] = \sum p_i \lceil \log(1/p_i) \rceil$

**Idealization**  $\forall i : |S(a_i)| = \log(1/p_i), E[S] = \sum p_i \log(1/p_i)$

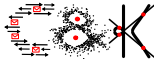


Given  $A = \{a_1, \dots, a_d\}$ , Shannon code  $S : A \rightarrow \{0, 1\}^*$  achieves

- 1  $\forall i : |S(a_i)| = \lceil \log(1/p_i) \rceil$
- 2  $E[S] = \sum p_i \lceil \log(1/p_i) \rceil$

**Idealization**  $\forall i : |S(a_i)| = \log(1/p_i), E[S] = \sum p_i \log(1/p_i)$

**Question** What happens, if we start with "wrong" distribution  $q$  to construct Shannon code  $S'$ ?



Given  $A = \{a_1, \dots, a_d\}$ , Shannon code  $S : A \rightarrow \{0, 1\}^*$  achieves

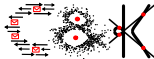
- 1  $\forall i : |S(a_i)| = \lceil \log(1/p_i) \rceil$
- 2  $E[S] = \sum p_i \lceil \log(1/p_i) \rceil$

**Idealization**  $\forall i : |S(a_i)| = \log(1/p_i), E[S] = \sum p_i \log(1/p_i)$

**Question** What happens, if we start with "wrong" distribution  $q$  to construct Shannon code  $S'$ ?

**Loss in compression:**  $E[S'] - E[S] = \sum p_i \log(p_i/q_i) = D_{KLD}(p, q)$ .

# Loss in compression

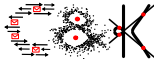


## Lemma 3.1

$$\forall p, q \in S^d : D_{KLD}(p, q) \geq 0.$$



# Loss in compression



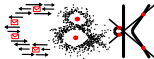
## Lemma 3.1

$$\forall p, q \in \mathcal{S}^d : D_{KLD}(p, q) \geq 0.$$

## Observation

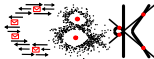
$$\forall x \in \mathbb{R}_+ : \ln(x) \leq x - 1.$$

# Markov models



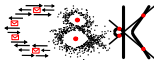
- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$

# Markov models



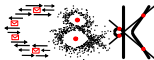
- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
- $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$

# Markov models



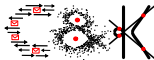
- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
- $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$
- $p_{cj} = \Pr[X_k = a_j | X_{k-1} \cdots X_{k-b} = c]$  for all  $k > b$

# Markov models



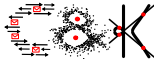
- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
- $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$
- $p_{cj} = \Pr[X_k = a_j | X_{k-1} \cdots X_{k-b} = c]$  for all  $k > b$
- for each  $c \in A^b$  an idealized Shannon code  $S_c$  for distribution  $P_c$

# Markov models



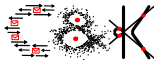
- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
- $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$
- $p_{cj} = \Pr[X_k = a_j | X_{k-1} \cdots X_{k-b} = c]$  for all  $k > b$
- for each  $c \in A^b$  an idealized Shannon code  $S_c$  for distribution  $P_c$
- use  $S_c$  to encode  $k$ -th symbol  $X_k$  if previous  $b$  symbols are  $c$

# Markov models



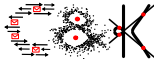
- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
- $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$
- $p_{cj} = \Pr[X_k = a_j | X_{k-1} \cdots X_{k-b} = c]$  for all  $k > b$
- for each  $c \in A^b$  an idealized Shannon code  $S_c$  for distribution  $P_c$
- use  $S_c$  to encode  $k$ -th symbol  $X_k$  if previous  $b$  symbols are  $c$
- first  $b$  symbols encoded somehow

# Markov models

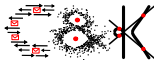


- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
  - $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$
  - $p_{cj} = \Pr[X_k = a_j | X_{k-1} \cdots X_{k-b} = c]$  for all  $k > b$
  - for each  $c \in A^b$  an idealized Shannon code  $S_c$  for distribution  $P_c$
  - use  $S_c$  to encode  $k$ -th symbol  $X_k$  if previous  $b$  symbols are  $c$
  - first  $b$  symbols encoded somehow
- + can yield very good compression if  $b$  is sufficiently large



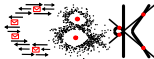


- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
  - $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$
  - $p_{cj} = \Pr[X_k = a_j | X_{k-1} \cdots X_{k-b} = c]$  for all  $k > b$
  - for each  $c \in A^b$  an idealized Shannon code  $S_c$  for distribution  $P_c$
  - use  $S_c$  to encode  $k$ -th symbol  $X_k$  if previous  $b$  symbols are  $c$
  - first  $b$  symbols encoded somehow
- + can yield very good compression if  $b$  is sufficiently large
- for large  $b$  have to store many ( $= |A|^b$ ) codes

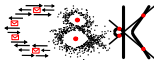


- $X = X_1 \cdots X_l$  sequence over alphabet  $A$ ,  $b \in \mathbb{N}$
  - $\forall c \in A^b$  : distribution  $P_c = (p_{c1}, \dots, p_{cd})$
  - $p_{cj} = \Pr[X_k = a_j | X_{k-1} \cdots X_{k-b} = c]$  for all  $k > b$
  - for each  $c \in A^b$  an idealized Shannon code  $S_c$  for distribution  $P_c$
  - use  $S_c$  to encode  $k$ -th symbol  $X_k$  if previous  $b$  symbols are  $c$
  - first  $b$  symbols encoded somehow
- + can yield very good compression if  $b$  is sufficiently large
- for large  $b$  have to store many ( $= |A|^b$ ) codes
- $\Rightarrow$  may outweigh gain of compression

## Compression and clustering

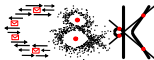


**Idea** Use large  $b$ , then use "few" ( $= k$ ) representative distributions to compress.



**Idea** Use large  $b$ , then use "few" ( $= k$ ) representative distributions to compress.

- $P = \{P_c | c \in A^b\}$
- find set of  $k$  centroid distributions  $\mathcal{C} = \{c_1, \dots, c_k\}$  and partition  $C_1, \dots, C_k$  of  $P$
- if  $P_c \in C_j$ , use idealized Shannon code  $S_j$  for distribution  $c_j$  instead of code for distribution  $P_c$

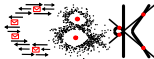


**Idea** Use large  $b$ , then use "few" ( $= k$ ) representative distributions to compress.

- $P = \{P_c | c \in A^b\}$
- find set of  $k$  centroid distributions  $\mathcal{C} = \{c_1, \dots, c_k\}$  and partition  $C_1, \dots, C_k$  of  $P$
- if  $P_c \in C_j$ , use idealized Shannon code  $S_j$  for distribution  $c_j$  instead of code for distribution  $P_c$

**Goal** Find centroids and corresponding partition that minimize loss in compression.

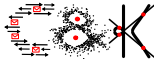
## Loss in compression



Loss is given by

$$\sum_{i=1}^k \sum_{P_j \in C_i} D_{KLD}(P_j, c_i)$$

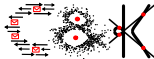
## Loss in compression



Loss is given by

$$\sum_{i=1}^k \sum_{P_j \in C_i} D_{KLD}(P_j, c_i) = \text{cost}^{KLD}(P, C)$$

## Loss in compression

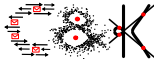


Loss is given by

$$\sum_{i=1}^k \sum_{P_j \in C_i} D_{KLD}(P_j, c_i) = \text{cost}^{KLD}(P, C)$$
$$\geq \text{cost}_k^{KLD}(P)$$



## Loss in compression



Loss is given by

$$\sum_{i=1}^k \sum_{P_j \in C_i} D_{KLD}(P_j, c_i) = \text{cost}^{KLD}(P, C)$$
$$\geq \text{cost}_k^{KLD}(P)$$

⇒  $k$ -median problem for Kullback-Leibler divergence