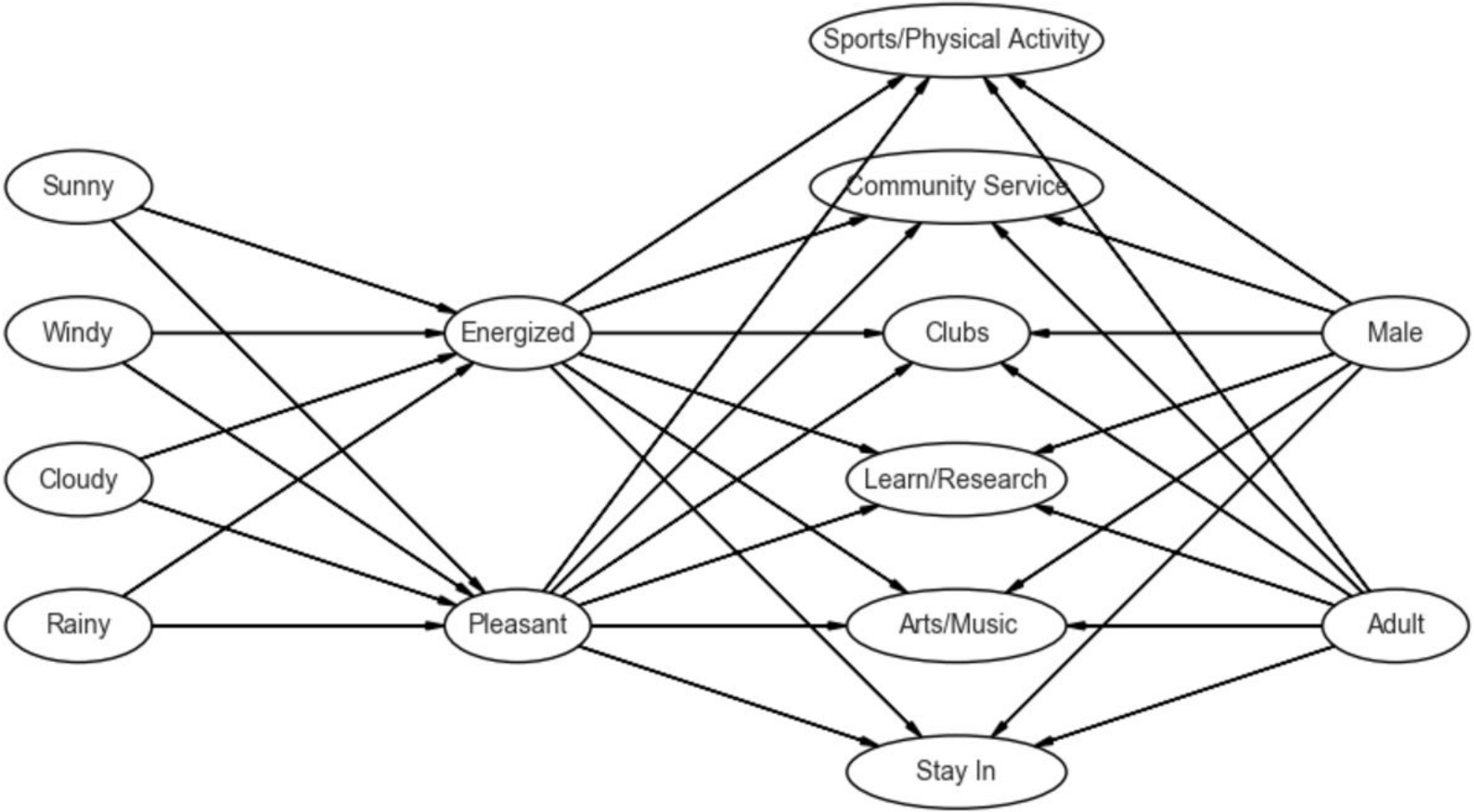
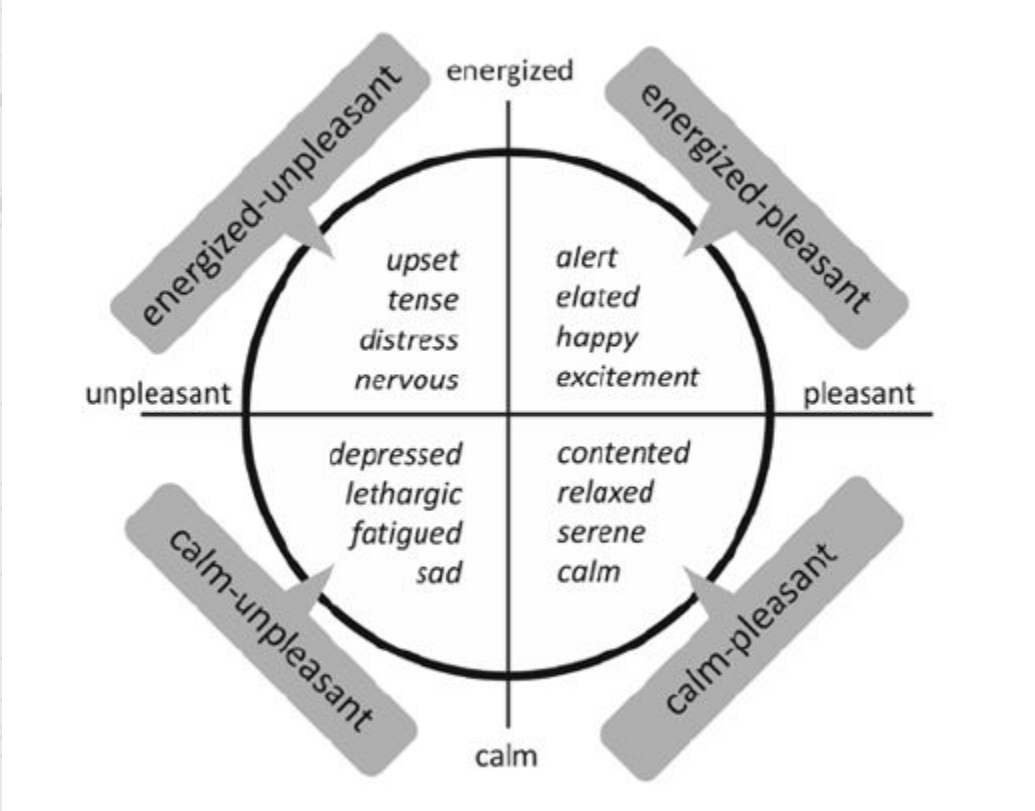
MCMC implementation

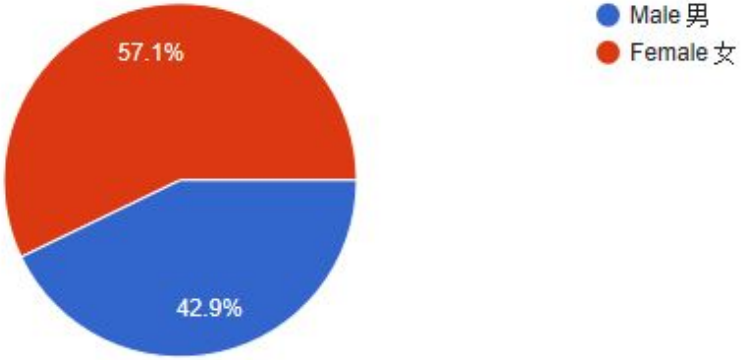
林文智, 亞琳娜, 蔡潔詩


Sports/Physical Activity
Sunny
Community Service
Windy
Energized
Clubs
Male
Cloudy
Learn/Research
Rainy
Pleasant
Arts/Music
Adult
Stay In

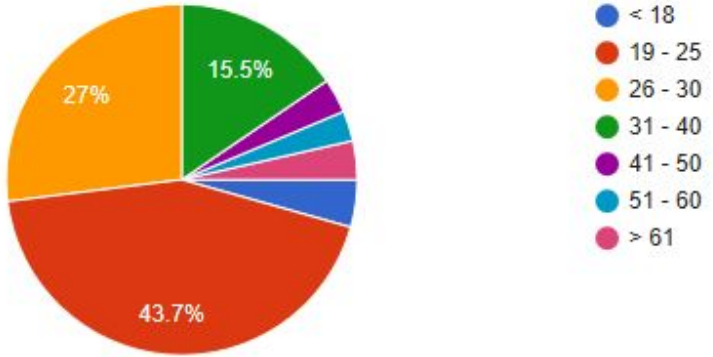


source: Four basic mood categories;
based on the PANAS model by Watson and Tellegen (1985),
with examples of moods (in the circle)
from Russell (1980) and Barrett & Russell (1999).

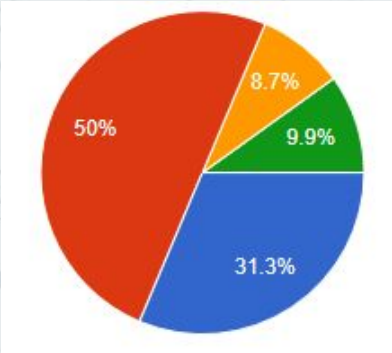
Gender 性别` (252 responses)



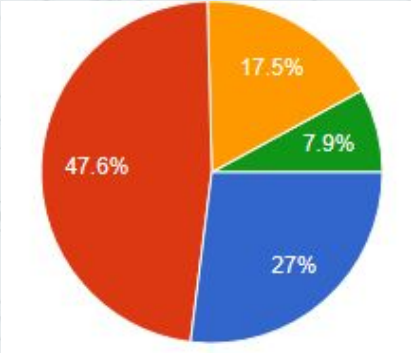
Age 年龄 (252 responses)



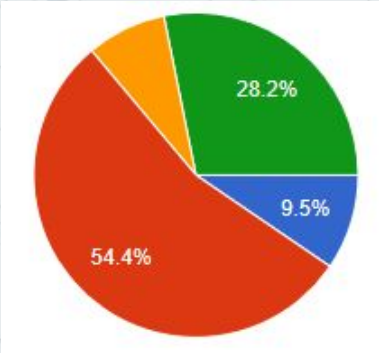
Which of the following best describe how you usually feel on...



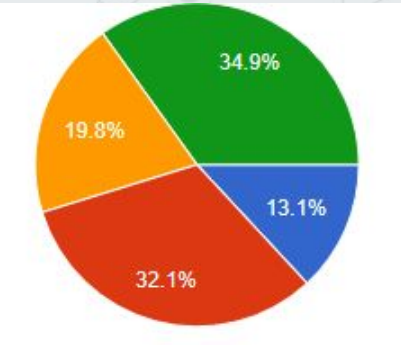
Sunny day



Windy day



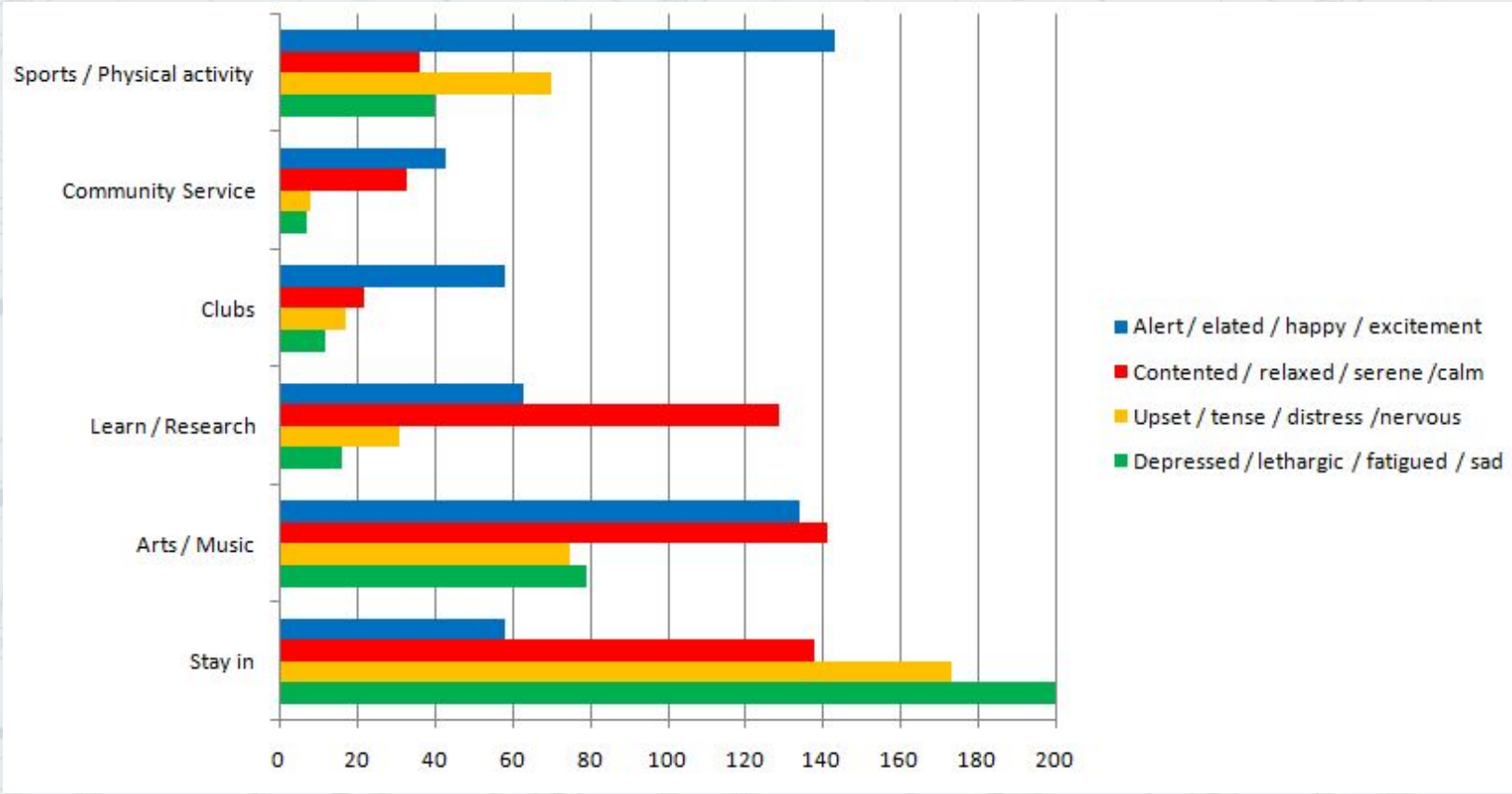
Cloudy day



Rainy day

- Alert / elated / happy / excitement
  警報/興奮/快樂/興奮
- Contented / relaxed / serene / calm
  滿足/放鬆/寧靜/平靜
- Upset / tense / distress / nervous
  生氣/緊張/苦惱/緊張
- Depressed / lethargic / fatigued / sad
  沮喪/嗜睡/疲勞/悲傷

Which of the following activities would you prefer to participate when you are feeling...



Probability Tables

Node:	Energized				
S	W	C	R	P(E)	P(~E)
T	T	T	T	0.007936507937	0.9920634921
T	T	T	F	0.0119047619	0.9880952381
T	T	F	T	0.02380952381	0.9761904762
T	T	F	F	0.09523809524	0.9047619048
T	F	T	T	0.0119047619	0.9880952381
T	F	T	F	0.03571428571	0.9642857143
T	F	F	T	0.05555555556	0.9444444444
T	F	F	F	0.1587301587	0.8412698413
F	T	T	T	0.02777777778	0.9722222222
F	T	T	F	0.02380952381	0.9761904762
F	T	F	T	0.09920634921	0.9007936508
F	T	F	F	0.1547619048	0.8452380952
F	F	T	T	0.03571428571	0.9642857143
F	F	T	F	0.01984126984	0.9801587302
F	F	F	T	0.06746031746	0.9325396825
F	F	F	F	0.1706349206	0.8293650794

Node:	Sunny
P(S)	P(~S)
0.2951807	0.7048193

Node:	Windy
P(W)	P(~W)
0.0813253	0.9186747

Node:	Cloudy
P(C)	P(~C)
0.8915663	0.1084337

Node:	Rainy
P(R)	P(~R)
0.6385542	0.3614458

Node:	Male
P(M)	P(~M)
0.571	0.429

Node:	Adult
P(A)	P(~A)
0.519	0.481

Gibbs sampling

function GIBBS-ASK($X, \mathbf{e}, bn, N$) **returns** an estimate of $\mathbf{P}(X|\mathbf{e})$
  local variables: $\mathbf{N}$, a vector of counts for each value of X, initially zero
    $\mathbf{Z}$, the nonevidence variables in bn
    $\mathbf{z}$, the current state of variables $\mathbf{Z}$, initially random

  for $i = 1$ to N **do**
    choose Z_j in $\mathbf{Z}$ uniformly at random
    set the value of Z_j in $\mathbf{z}$ by sampling from $\mathbf{P}(Z_j|mb(Z_j))$
    $\mathbf{N}[x] \leftarrow \mathbf{N}[x] + 1$ where x is the value of X in $\mathbf{z}$
  return NORMALIZE($\mathbf{N}$)

source: Stuart Russell, CS professor in Berkeley,
Bayesian networks: Inference and learning
CS194-10 Fall 2011 Lecture 22

Markov Blanket

$$P(x'_j|mb(X_j)) = \quad P(x'_j|parents(X_j))\prod_{Z_\ell \in Children(X_j)} P(z_\ell|parents(Z_\ell))$$

source: Stuart Russell, CS professor in Berkeley,
Bayesian networks: Inference and learning
CS194-10 Fall 2011 Lecture 22

```
def gibbs_ask(self,x,e,bn,N):
    #returns an estimate of P(x|e)
    #prepare graph
    result = [0,0]
    bnNew = copy.deepcopy(bn)
    mNodes = bnNew.fill_nodes(e)
    myX = bnNew.get_x_node(x)
    for i in range(N):
        shufList = [i for i in range(len(mNodes))]
        shuffle(shufList)
        for z in shufList:
            ##Set value of Zj in z by sampling from P(Zj|mb(Zj))
            zState = self.mb(mNodes[z],bnNew)
            bnNew.set_state(zState,mNodes[z])
            xState = self.mb(myX,bnNew)
            if xState:
                result[1] = result[1] + 1
            else:
                result[0] = result[0] + 1

    total = result[0] + result[1]
    normalizedFalse = result[0]/total
    normalizedTrue = result[1]/total
    return {True: normalizedTrue, False: normalizedFalse}
```

```
def mb(self,z,bn):
    #returns probability of Z in the markov blanket space in BN
    #P(x given mb(X)) = P(x given parents(X)) multiplied by the multiplication
    # of all children Z with formula P(Z given parents(Z))
    probParent = bn.get_prob(z)
    probChild =  bn.get_prob_children(z)
    return bn.def_new_state(probParent*probChild, z)
```

Result Analysis:

Based on weather only

probability of 'energized' given cloudy=True, sunny=False, windy=True, rainy=False, is: {False: 0.8359, True: 0.1641}

probability of 'pleasant' given cloudy=True, sunny=False, windy=True, rainy=False, is: {False: 0.9003, True: 0.0997}

… adding opposite mood

probability of 'energized' given cloudy=True, sunny=False, windy=True, rainy=False, pleasant=False, is: {False: 0.8684444444444445, True: 0.13155555555555556}

probability of 'pleasant' given cloudy=True, sunny=False, windy=True, energized=False, rainy=False, is: {False: 0.8746666666666667, True: 0.12533333333333332}

… adding activity

probability of 'energized' given cloudy=True, rainy=False, pleasant=False, sunny=False, windy=True, sport=True, is:

{False: 0.51425, True: 0.48575}

probability of 'pleasant' given cloudy=True, rainy=False, energized=False, sunny=False, windy=True, sport=True, is:

{False: 0.503125, True: 0.496875}

… adding age

probability of 'energized' given cloudy=True, rainy=False, pleasant=False, sunny=False, windy=True, sunny=True, adult=True, is: {False: 0.4797142857142857, True: 0.5202857142857142}

probability of 'pleasant' given cloudy=True, rainy=False, energized=False, sunny=False, windy=True, sunny=True, adult=True, is: {False: 0.5032857142857143, True: 0.4967142857142857}t