# 第26回数理科学コンクール課題解説

令和5年11月3日 千葉大学先進科学センター

# 目 次


**1 はじめに** **2**

**2 優秀者氏名** **4**

**3 課題 1** **7**

解説 . . . . . 9

講評 . . . . . 10

**4 課題 2** **11**

解説 . . . . . 13

講評 . . . . . 14

**5 課題 3** **15**

解説 . . . . . 16

講評 . . . . . 17

**6 課題 4A** **18**

解説 . . . . . 19

講評 . . . . . 20

**7 課題 4B** **21**

解説 . . . . . 26

講評 . . . . . 31

**8 ロボットの部** **33**

解説 . . . . . 34

講評 . . . . . 34

**9 人工知能の部** **36**

解説 . . . . . 40

講評 . . . . . 43

# 1　はじめに

明治の文明開化以来, 我が国は欧米先進国の科学技術を効率よく吸収して発展してきました. 戦後もこの傾向は基本的には変わっていません. 現在, 我が国は大量の自動車や電子機器を輸出して経済大国となっていますが, これらの工業製品の基本原理はほとんど外国で考えられたものです. 欧米諸国との間に経済摩擦や文化摩擦が生じている現状を考えると, これからの我が国で大切なことは独創性のある個性的な人材を育成して, 新しい科学技術のフロンティアを切り開き, 世界に貢献することであると考えられます.

千葉大学では, 日本のみならず, 世界の科学技術の先端を担う若者を発掘し, 育成するための一助として, 本年度も, 第 26 回数理科学コンクールを開催しました.

本年度から, 対面開催でのコンクールを復活しました. それに合わせて, 今回から「ロボットの部」も復活しました. 過去 3 回の経験から, 課題の部は, 遠隔開催も引き続き実施することとしました. そして, 本年度より「人工知能の部」を開催します. 人工知能の部は完全に遠隔開催としました.

AI, データ・サイエンスに関する報道は落ち着きを見せてきています. AI, データ・サイエンスの手法がいろいろな現場で普通の考え方になってきました, また, AI, データ・サイエンスの課題を取り扱い, 解決するためのプログラムも種々公開せれています, そこで, 千葉大学数理科学コンクールでは, 本年度より「人工知能の部」を開催しました.

数理科学コンクールの主題である, 「自ら実験をして現象を考察する. 」を引き続き実施するために, 遠隔開催参加者には実験機材を送ることにしました.　機材の準備には, 千葉大学大学院の学生の皆さんにも協力してもらいました. 配送できる機材の大きさや種類には制限があるため, 皆さんが受け取る機材の中には自分の使いたいものが入っていないかもしれませんでした. しかし, 与えられた機材だけを使って, 実験の方法を考察して工夫することも科学者にとって重要な訓練です. 今回から, 対面開催の参加者と遠隔開催の参加者と共通の課題対面参加者のための課題, 遠隔参加者のための課題, 3 種類の課題を用意しました. それぞれ, 4 つの課題に取り組んでもらいました. 対面開催への参加者は従来通りに, コンクール当日の提出締切時間までに, 答案を提出してもらいました. また, 遠隔開催への参加者は, 開催案内に書かれている期日までに解答を千葉大学に返送してもらいました.

ロボットの部は, 第 1 日目にロボットの動作に関するプログラムの講習を受けてもらいました. 第 2 日目 (対面開催課題の部実施日) にプログラムを作成し, 第 2 日目の午後 3 時以降に動作評価のためのコンテストを実施した.

人工知能の部は, 課題を別途, 登録した住所に送り, 宛てのメールに添付して提出してもらいました.

このコンクールの特色は次の通りです.

1. 自由にゆったり考える

   試験時間は 1 日, 会場や自宅で時間を自由に使い, 解答を導く. また, インターネットの検索も条件を付けて可としました.

2. たのしい物理・数学の発見

   物理や数学のカリキュラムにとらわれず, 物理や数学の本質に根ざした, 考えて楽しい問題を提供する.

3. 多彩な才能の評価

様々な参加者の優秀な能力やユニークな発想を多面的に評価するため, 問題をたくさん解いたものだけでなく, 1 題に集中してすばらしい発想を出したものも表彰の対象にする.

.

4. 人材の育成

コンクール参加者の物理や数学の能力をさらに高めるため, ホームページ上に表彰者の名前と講評を掲載する.

過去 25 回のコンクールに引き続き, 多くの中高生の参加者があり, いろいろユニークなアイディアが生まれました. 中学生も, 高校生に負けず優秀でありました. そして, 答案を見ると, それぞれの問題に興味を持ちながら解答していることが読んでとれました.

第 26 回数理科学コンクールの課題の解説と提出された答案の評価を以下にまとめます. 解説に述べてあるように, 各課題は課題出題者の周りにある基本的な問題や最先端の問題, さらには歴史的に意味のある問題を元にして作成しました. 課題提出者一同, みなさんの素晴らしい洞察力と表現力を前にして, 大変感心いたしました.

参加者の皆さんが今後, 科学する心を磨き続け, 我国の科学の発展に貢献することを課題作成者一同希望します. 今後も諸君と共に科学することを楽しみたいと考えています. 千葉大学では今後も引き続きこのコンクールを実施する予定です. 物理・数学に興味がある中高生の積極的な参加を期待しています. 課題作成者もさらに研鑽をかさね, おもしろく, しかも科学の本質に迫る課題を考えていきます.

課題作成者

千葉大学教授　石井久夫
千葉大学名誉教授　井宮　淳
東京慈恵会医科大学教授　植田　毅
(五十音順)

令和 5 年 11 月 3 日

## 2　優秀者氏名

令和 5 年 7 月 17 日に開催した第 26 回数理科学コンクール課題の部, 7 月 16 日と 17 日に開催した第 26 回数理科学コンクールロボットの部, 8 月 27 日に開催した第 26 回数理科学コンクール人工知能の部, それぞれの参加者の皆さんのすばらしい答案の中から以下の参加者諸君を表彰することを決定しました.

**第 26 回数理科学コンクール優秀者**

**課題の部**

| | | | |
|---|---|---|---|
| 金欅賞 | 北村 涼太 | 大場 葉月 | 北岡 楓眞 |
| | 佐久間 崇太 | 酒井 大和嘉 | 田沼 佑騎 |
| | 田中 喜大 | 伊藤 和生 | |
| | 土屋 晴奈 | 長山 茉生 | 米原 百香 |
| 銀欅賞 | 伊藤 真咲 | 熱田 瑞季 | |
| | 吉澤 汀子 | | |
| | 竹味 真俐 | 福田 琥太郎 | 小島 星司 |
| | 堀 真奈美 | 大場 杜輔 | 藤井 充希 |
| | 澁谷 奏良 | 上田 宜佳 | 中村 郁雅 |
| | 神取 雄大 | 坂本 悠月 | 室井 海灼 |
| | 大中 洋奈 | 村川 博美 | |
| 学長賞 | 中村 颯人 | | |
| | 小林 杏珠 | | |
| | 関弥 建伸 | 鈴木 翔太 | |

**ロボットの部**

| | | | |
|---|---|---|---|
| 機巧賞 | 桂田 進次郎 | | |
| | 山﨑 隆人 | | |
| | 葛西 匠 | | |
| 敢闘賞 | 浅賀 慎之助 | 飯島 一翔 | 齋藤 慧翔 |

**人工知能の部**

| | | | |
|---|---|---|---|
| 知巧賞 | 林 哲彦 | 伊藤 康真 | |
| | 中村 哲 | 井上 偉貴 | 三田 悠人 |
| 学長賞 | 清水 陵佑 | 國吉 仁志 | 原田 悠世 |

| 課題 | 参加者氏名 | | |
|---|---|---|---|
| 1 | 堀 真奈美 | 大場 杜輔 | 藤井 充希 |
| | 佐久間 崇太 | 酒井 大和嘉 | 田沼 佑騎 |
| | 田中 喜大 | 伊藤 和生 | |
| | 土屋 晴奈 | 長山 茉生 | 米原 百香 |
| 2 | 中村 颯人 | | |
| | 佐久間 崇太 | 酒井 大和嘉 | 田沼 佑騎 |
| | 土屋 晴奈 | 長山 茉生 | 米原百香 |
| 3 | 小林 杏珠 | | |
| | 中村 颯人 | | |
| | 伊藤 真咲 | 熱田 瑞季 | |
| | 北村 涼太 | 大場 葉月 | 北岡 楓眞 |
| | 吉澤 汀子 | | |
| | 竹味 真俐 | 福田 琥太郎 | 小島 星司 |
| | 関弥 建伸 | 鈴木 翔太 | |
| | 佐久間 崇太 | 酒井 大和嘉 | 田沼 佑騎 |
| | 澁谷 奏良 | 上田 宜佳 | 中村 郁雅 |
| | 神取 雄大 | 坂本 悠月 | 室井 海灼 |
| | 大中 洋奈 | 村川 博美 | |
| 4 | 北村 涼太 | 大場 葉月 | 北岡 楓眞 |
| | 田中 喜大 | 伊藤 和生 | |

事前に了承を得た方の氏名のみ掲載しております。

千葉大学先進科学センター
センター長 教授　松浦彰
千葉大学大学院融合理工学府
数学情報科学専攻情報科学コース
コース長 教授　塩田茂雄

# 課題の部

## 3　課題 1

(共通課題)

作図問題と折り紙の関係を幾何学の問題として数学的に考えてみよう. 平面幾何学の中で定規 (目盛がなく, 直線を引くことができる器具) とコンパスとを使って作図する問題では, 以下の 5 つの操作

1. 与えられたの 2 点を通る直線を作図できる.
2. 与えられたの 1 点を中心とし, それ以外与えられた 1 点を通る円を作図できる.
3. 互いに平行でないの 2 直線の交点を作図できる.
4. 与えられた円と直線との高々2 個の交点を作図できる.
5. 与えられた 2 円の高々2 個の交点を作図できる.

を有限回施して作図できる図形, 又は描いた図形から得られる量 (辺の長さ等) を解とすることです. この操作は, 代数的には 2 次方程式を有限回解いてその解を求めることに相当します. また, 定規とコンパスで可能な作図において, すべての新しい点は,

- 2 つの円の交点
- 円と直線の交点
- 2 つの直線の交点

のいずれかです. 従って, 定規とコンパスとを使って作図する問題は上の 3 つの操作を有限回施して作図できる図形, 又は描いた図形から得られる量 (辺の長さ等) を解とすることです. 定規とコンパスで可能な作図においての基礎となるのはユークリッド幾何学の 5 つの公準

**第 1 公準**　与えられたの 2 点を直線で結ぶことが可能である.

**第 2 公準**　与えられたの 2 点を結んだ線分は両側に延長して直線にできる

**第 3 公準**　与えられた 1 点を中心にして任意の半径の円を描画可能である

**第 4 公準**　全ての直角は等しい（角度である）.

**第 5 公準**　1 つの直線と直交する 2 線分を延長しても交わらない. (平行線公準)

に基づく操作を定規とコンパスで実現したことと考えることができます. 第 5 公準から, 1 つの直線が 2 つの直線に交わり, 同じ側の内角の和が 2 つの直角より小さいならば, この 2 つの直線は限りなく延長されると, 2 つの直角より小さい角のある側において交わるであることが分かります.

一方, 紙の折り方に関して

1. 2 点 $p_1$, $p_2$ が与えられたとき, 2 点を通るただ 1 つの折り方がある.
2. 2 点 $p_1$, $p_2$ が与えられたとき, $p_1$ を $p_2$ に重ねるただ 1 つの折り方がある.

3. 2 直線 $l_1, l_2$ が与えられたとき, $l_1$ を $l_2$ に重ねるような折り方がある.

4. 1 点 $p_1$ と 1 直線 $l_1$ が与えられたとき, $l_1$ に垂直で $p_1$ を通るただ 1 つの折り方がある.

5. 2 点 $p_1, p_2$ と 1 直線 $l_1$ が与えられたとき, $p_1$ を $l_1$ 上に重ね, $p_2$ を通る折り方がある.

6. 2 点 $p_1, p_2$, 2 本の直線 $l_1, l_2$ が与えられたとき, $p_1$ を $l_1$ 上に重ね, かつ, p2 を l2 上に重ねる折り方がある.

7. 1 点 $p_1$ と 2 直線 $l_1, l_2$ が与えられたとき, $p_1$ を $l_1$ に重ね, $l_2$ に垂直な折り方がある.

の 7 つの性質は, 折り紙公理とよばれています.

ユークリッドの公準に基づく作図操作と折り紙公理との関係を数学における公理の立場から比較して, 作図操作と折り紙公理の相違点や類似点を数学的にまとめてください. ただし, 数学における公理とは

「ある体系を構築する場合に, 導入される最も基本的な仮定.」

の集合です. また, 幾何学の 5 つの公準は, 幾何学の問題を考える上で, 公理として採用されます. 詳しくは, ユークリッド幾何学の公理ともよばれます. 第 5 公準を見直して, 平行線が交わることがあるすれば, ユークリッド幾何学ではない幾何学として, 非ユークリッド幾何学が構成されます.

## 解説

計算機科学 (Computing Science) では, 処理 (Algorithm) を計算と捉え, ある処理を実行する手間を考察し, 手間によってアルゴリズムのクラス分けを行います. 計算の手間は計算複雑性と呼ばれます. 数値を計算する問題では, 主にある問題を計算する乗算の回数を評価します.

一方, 幾何学的な問題である作図問題では, 基本操作の操作回数を, 問題の計算複雑性として採用することがあります.

さて, 課題は折り紙公理の性質について聞いています. 折り紙公理は, 折り紙で実現できる操作を公理としています. その意味で, 基本操作集です.

平面幾何学に関して, ユークリッドは, 5 つのユークリッド公準

1. 任意の一点から他の一点に対して直線を引くこと
2. 有限の直線を連続的にまっすぐ延長すること
3. 任意の中心と半径で円を描くこと
4. すべての直角は互いに等しいこと
5. 直線が 2 直線と交わるとき, 同じ側の内角の和が 2 直角より小さい場合, その 2 直線が限りなく延長されたとき, 内角の和が 2 直角より小さい側で交わる

を認めています. この公準をユークリッドの平面幾何学の公理と呼ぶことがあります. 課題では, この公準を平面幾何学の公理と呼ぶ立場をとりました.

加えて, 次の 9 個の公理がユークリッドによって示されています.

1. 同じものに等しいものは, 互いに等しい.
2. 同じものに同じものを加えた場合, その合計は等しい.
3. 同じものから同じものを引いた場合, 残りは等しい.
4. 不等なものに同じものを加えた場合, その合計は不等である.
5. 同じものの 2 倍は, 互いに等しい.
6. 同じものの半分は, 互いに等しい.
7. 互いに重なり合うものは, 互いに等しい.
8. 全体は, 部分より大きい.
9. 2 線分は面積を囲まない.

ユークリッドの公準は, 幾何学的作図で実現できる性質を記述しています. 平面幾何学の操作集と捉えることもできます. 定規とコンパス (定規に対応して, 円規と言われてこともあるそうです. ) による操作集と捉えることもできます.

数学的には, 5 公準を利用して平面上に作図できる性質は, 2 次方程式を解くことに帰着されることも知られています. ユークリッドの公理の第 9 公理以外は, 全ての学問の根源になる概念です.

従って, 折り紙公理は, ユークリッドの平面幾何学に対する公準に対応する記述と考えることもできます. 5 公準の内, 第 1 公準から第 3 公準までを受け入れれば, 定規とコンパスによる作図が可能と成ります. 従って, 第 1 公準から第 3 公準を基本操作と考え, 作図問題の計算複雑性を定義することができます.

## 講評

幾何学的操作として, 論考した解答はありませんでした.

## 4　課題 2

(共通課題)

ガソリンエンジンは内燃機関の 1 種であり, その代表的なものです. ここでは, エンジンはレシプロエンジンと呼ばれ, 爆発から回転運動を取り出す装置です. 世界的に環境保護の立場から, ガソリンエンジン, ディーゼルエンジンなどの内燃機関を搭載した自動車を電気自動車に置き換えていく動きが加速しています. しかし, 極寒・極暑で, 人里から離れた場所では, 現在までの運用実績による信頼性から自動車だけではなく, 発電機などにも今なお, レシプロエンジンの利用されます. また, 船舶の駆動系も同様です. この状態はしばらく解決できないかもしれません.

内燃機関では, 機械内部で発生させる爆発から回転運動をとりだしています. その方法の概要は閉じた空間で爆発を起こし, その爆発から発生する膨張による平行運動を回転に変えます. このようなエンジンをレシプロエンジンといいます爆発を起こす場所を気筒 (シリンダー) といいます. レシプロエンジンの開発の歴史は, 気筒を増やすことと, その配置を変えることです. レシプロエンジンの性質性能を評価する指標の 1 つが, 気筒数です. エンジンの気筒数は 1 気筒, 2 気筒, 3 気筒, 4 気筒, 5 気筒, 6 気筒また, 気筒の配置は, 直列, V 字, 水平対向などがあります. 水平対向とは気筒を水平に配置し, 回転軸の両側からピストンを往復させる形式エンジンです. 水平対向 6 気筒エンジンを搭載した車種を製造・販売する有名な自動車メーカーの 1 つがポルシェです. さて, レシプロエンジンが爆発によるピストンの往復運動を回転運動に変換することから, 装置全体に振動が発生します. 自動車設計者, 航空機設計者にとって一番の問題は, ピストンの上下運動から発生する振動を抑えることです. そのために, 気筒数を増やし続けたとも言えます. 気筒ないでの爆発を連続的に動作さる機構の 1 つが, 4 サイクルエンジンでう. 気筒の燃焼室内で, 燃料の爆発を起こすために, 4 つの過程「吸気・圧縮・爆発燃焼・排気」を行います. 気筒数とエンジン本体の振動の関係を解析し, 振動を抑えるのに適した気筒数を解析してください. ピストンの往復運動は最上位点, 最下位点で瞬間的に停止します. この点をそれぞれ, 上死点, 下死点とよびます. 2 つの死点から, ピストンが反対方向に動き出すことから偶力による振動が発生します. この振動をレシプロエンジンの 1 次振動とよびます. この振動によって, クランク 1 回転間にエンジン本体が 1 回上下に振動します. ピストンが死点から運動方向を変えるときに気筒（シリンダー）の内壁に力がかかります. 回転する軸にクランクで結ばれたシリンダーヘッドが内壁に掛ける力の方向は上死点, 下死点で反対方向に成ります. 従って, シリンダーヘッドがシリンダー内壁を押し付ける偶力により, 回転軸 1 回転の間に 2 回振動が発生します. この振動を 2 次振動と呼びます. それぞれの気筒から, 1 次振動, 2 次振動が発生します. これ以上の次数の振動も発生します. しかし, 影響が小さいので, 気筒数, 気筒配置, 死点位置を組み合せて, 1 次振動, 2 次振動を抑えることが問題になります. 死点位置の組合せとは, 軸が回転中にピストン間で死点が角度でどれだけずれかるかの設定です. 例えば, 2 気筒のレシプロエンジンで, ピストン 1 が上死点にあるとき, ピストン 2 が下死点にあるとき, 死点配置は 180 度ずれているといいます. 以下では 4 サイクルのレシプロエンジンを考えます.

**問 1** (対面課題) 直列 $n$ 気筒のエンジンのピストンに, 端から順に, 1 から $n$ までの番号を付けます. 1 次振動, 2 次振動を最小にする死点の配置を求めて下さい.

**問 2** 水平対向 $2n$ 気筒のエンジンのピストンに, 端から順に, 1 から $n$ までの番号を付けます. 1 次振

動, 2 次振動を最小にする死点の配置を求めて下さい.

　レシプロエンジンより振動の少ないエンジンとしてロータリーエンジンが知られています. ロータリーエンジンはピストンの上下による振動の発生しない星形 3 気筒エンジンと考えられます. しかし, その構造から回転軸が擂粉木運動をするため, 違った機構でエンジンに振動が発生します.

**問 3** ロータリーエンジンの軸の擂粉木運動を最小にする方法を考えてください.

## 解説

環境保護への取り組みから, 電気自動車への移行が進行しています. しかし, 内燃機関は熱エネルギーから運動エネルギーを取り出す移動可能なエネルギー変換装置として, 完全には電動機に置き換わることが現在でも想定はできません. 船舶用の動力源は依然, C 重油を燃料とする 2 気筒ディーゼルエンジンが主流です.

種々の社会情勢から, 温暖地では, 移動用動力を内燃機関から電気動力に切り替えていくことは可能でしょう. 極寒地, 極熱帯地では, 電池の給電能力が低下することから, 内燃機関に頼る必要があります. また, 電気を送電系統のない場所で利用するためには発電機が必要です. 移動型発電機の動力として利用される装置も内燃機関です. 海外との荷物や物資の輸送には, 海を越えて船舶で行うことが現在のところ効率的です. 先に述べた, 船舶用ディーゼルエンジンは, 1 分間に 60 回転程度を行います. 自動車のエンジンの回転数と比べると, 相当ゆっくりとしています. 一方, 1 航海, 1ヵ月以上の間回転を続ける必要があることから, 身の回りにある自動車のエンジンとその設計思想が大幅に異成ります. また, 自動車のガソリンエンジンの熱交換率が 35%程度であることに対して, 船舶用エンジンでは 50%を超えるものもあります.

現在では, 動力を外燃機関 (蒸気ピストン) に頼ることはめったにありません. 蒸気機関車は観光用に保存され, 一部, 新造されています. 1940 年代から 1950 年代にかけて, 蒸気機関車に対して種々の可能性が試されました. 例えば, 復水型にして蒸気を水に戻して再利用することによって給水間隔を長くする試み, 超高圧蒸気を発生させて熱交換率を上げる試み, などです. しかし, 当時の加工技術力では加工精度を上げることができずに, 長期に利用されることはありませんでした. しかし, 現在これらの技術は再評価されて, 環境保全型の観光鉄道などに利用されています. このような背景から, 内燃機関に関する研究を捨て去ることなく続ける必要があります.

一方, 電池を動力源とすることによって, 車体の重量が増えます. このことから, 道路などの交通施設の設計思想も現在の内燃機関移動車のものから変更する必要もあります.

自動車用内燃機関に加えて, 航空機用内燃機関も重要です. 自動車用エンジンの効率を上げる加圧気 (ターボ) に関する研究も重要です. 同じ原理の加圧気が, 高能率なジェットエンジンにも必要です.

このような理由から, エンジンの課題を出題しました.

エンジンの振動に関する解析は, 自動車の出現から研究されてきました. 現在では, 機械工学の一部です. 内燃機関の振動に係る研究は, 高性能な航空機用エンジンの開発への必要性から, 航空機が戦争で多数使用された第一次世界大戦, 戦間期を経て, ジェットエンジンが実用に耐えるようなるまで活発に続けられてきました. 基本的な性質の網羅的解明は, 1940 年代に完成したとされています. 1934 に書版出版された Jacob Pieter Den Hartog の Mechanical Vibrations がこの分野の古典的名著です. 日本語にも訳されています. 版を重ねて 2008 年の復刻版があります. 現在でも, 機械振動の良い教科書として参考に成ります.

発火点の調整が必要でが, 直列 6 気筒 4 サイクルエンジンが最も振動を抑制することが, 上記 Mechanical Vibrations に記載されています. このことから,4 サイクルエンジンの場合直列 6n 気筒エンジンが振動を抑制することが分かります. 死点を 120° づつずらした、直列 3 気筒 4 サイクルエンジンでも振動が抑えられます. しかし, 回転軸に擂粉木運動が発生します. 擂粉木運動を抑制するために,

擂粉木運動を打ち消す 3 気筒分を追加すれば, 擂粉木運動が 180∘ ずれていればこの制御が可能です.

従って、3 つ気筒の中で燃料が燃焼する順を逆にして, 倍の 6 気筒にすることします. . 直列 $6(= 2\times3)$ 気筒 4 サイクルエンジンが振動をうち消すことに成ります. さらに, $n\times(2)$ にして, 6 気筒の部分を繋げれば, 大きな出力のエンジンを設計できます. 点火順序は通常, $1 \to 5 \to 3 \to 6 \to 2 \to 4$ です. これは, $A = (1, 2, 3)$ と $B = (4, 5, 6)$ とを 3 気筒の組と考えます 1 を真上にし, 2,3, を時計回りに, 120° ずらせて配置します. また, 6 を真上にして, 5,4 を反時計周りに 120° ずらせて配置しますこのダイアグラムの番号を, $A$ の要素, $B$ の要素となるように交互に順にたどる列 $a_1 \to b_1 \to a_2 \to b_2 \to a_3 \to b_3 \to a_1 \to$ が $1 \to 5 \to 3 \to 6 \to 2 \to 4$ に成ります. 死点は位置, 発火点配置の逆に成ります.

船舶用ディーゼルエンジンは 2 サイクルですが, 大型商船では, 18 気筒が使われます. 気筒 1 つの大きさは人間 1 人が中に入る程度の大きさです.

## 講評

自動車整備士は, エンジンの発火点を調整して, エンジンを最良の状態に整備する必要があります. 自動車整備士は, 自動車整備士技能検定に合格し, 資格を得る必要があります. インターネット上に, エンジンの動作説明に関する説明があります. 自動車ファンが少なからず現在も存在するのでしょう.

インターネット上の記事を参考にした解答でも, 記事の内容を理解している解答は評価しました. また, ロータリーエンジンとの関係を論じた解答は, 独創的でした.

# 5　課題 3

(共通課題)

この世では, 一般に, あるもの（台）の上に置いてある物体を動かす (運動させる) ためには, ある大きさ以上の力を加える必要があります. これは,「物体が力を受けると, その力の方向 · 向きに加速度を生じ, その加速度の大きさは力に比例し, 物体の質量に逆比例する.」というニュートンの運動の第 2 法則に反しているように見えます. これは, 経験上よく知られているように, 台と物体の接触面に摩擦力がはたらいているからです.

摩擦現象について, 観察に基づく経験的な摩擦の基本法則 (アモントン・クーロンの法則)) があります. その法則は次の 4 つの項目から成ります.

1. 摩擦力は垂直荷重に比例する.
2. 摩擦力は見かけの接触面積には無関係である.
3. 運動摩擦の摩擦力は, 滑り速度には無関係である.
4. 静止摩擦力は運動摩擦力よりも大きい.

摩擦の法則を検証してください ( 4 項のうちの一つでもよい.). 特に, 何故「見かけの接触面積」に無関係なのか, 摩擦の原因とは何かについて考察してください. 検証は実験的にでも理論的にでも構いません.

実験を行う場合は, 以下の項目を考え, 実験計画を明示し, 測定結果は表にまとめたり, グラフを描いて考察してください.

- 何を検証するのか（実験の目的）
- どのような道具, 器具を用いるのか
- どのような物理量を何を用いて測定するのか
- 測定結果（表, グラフ）
- その測定結果からどのようなことが言えるのか考察する

## 解説

この世で生活する限り摩擦から逃れることはできません. どのような物質の間でも摩擦は存在します. エジプトのピラミッド建設, 産業革命以後の機械の開発などは人類と摩擦の戦いと言っても過言ではありません. 生物の体内でもその物理法則から免れることはできず, 骨と骨の摩擦を軽減するために軟骨を発達させました. スケーリング則的には人類の大きさの動物の寿命はおよそ 50 歳ということになりますが, 人類はそれ以上の長寿を実現し, 軟骨のすり減りによる痛みに悩まされることになっています. また, 逆に, 摩擦がなければテニス, 卓球, 野球, ボーリング, ゴルフなどのボールを用いたスポーツは面白い競技になってないでしょう.

摩擦の法則性について最初に記録に残したのは, 彼のレオナルド・ダ・ビンチであると言われています. 摩擦の法則についての実験装置のスケッチも残しています. その約 200 年後, 1699 年にフランスのアモントンが力学的な摩擦の法則を発表しています. 内容的にはクーロンがまとめた法則と（異なっている部分もありますが）実質的には同じでしたが, 産業革命により機械の改良が切実となっていた時代背景によりクーロンの研究が広く認知されることなりました.

摩擦の法則は日常で出会い, 高校物理でも扱う身近な法則ですが, 微視的なモデルからきっちりとした理論が構築され, 物理定数により摩擦係数を表すようなことは未だにできていません. その意味で, 経験則から抜け出せていません.

高校の物理では一大法則のように教えられる摩擦の法則ですが, 物質同士の接触面の表面状態は未だに精力的に研究されています. 例えば, 超真空中で平坦な金属の表面にプラズマを当てて, 酸化膜を取り除いたもの同士を接触させると, もともと一体の金属だったかのように融合してしまいます. 通常, 摩擦と言えば滑り摩擦のことですが, それとは別に, 転がり摩擦というものがあります. ものを円周に沿って回転させるときの摩擦ですが, 滑り摩擦と同じように摩擦係数を定義でき, おおよそ滑り摩擦係数の 1/100 程度になる. ただ, その摩擦係数は滑り摩擦のものとは本来の意味としては異なる.

それでも, 歩行や自転車, 自動車などの加速について考えると, どちらも摩擦力の反作用を用いて加速します.

さて, 摩擦の法則が物理の基本法則であればそれを受け入れるしかありませんが, そうでないならば, 基本法則からそれを説明しなければなりません. この課題ではそれに挑戦してもらったわけです.

クーロン自身は, 固体表面には細かな凸凹があり, その凸凹を乗り越えて細かい上下動を繰り返しながら滑るというイメージを抱いていいたようですが, そのようなイメージでは凸凹がかみ合ってしまった場合動かなくなることが考えられます. 力によって, その凸凹も互いに変形する効果なども考慮しなければなりません. そこで出てくるのが, 変形する凸凹した表面で実際に接触している面積, これを見かけの接触面積に対し, 真実接触面積と言います. 真実接触面積は面にはたらく荷重に比例するとして, 摩擦力はこの真実接触面積に比例すると考えると, 摩擦力は荷重に比例することに成ります.

このとき, 真実接触面でどのような力がはたらいていて摩擦力を発生させているかということは, 摩擦の法則の説明には避けて通れません. 現在では, 分子間力が真実接触面にはたらいていると考えられています. 分子間力は分子性結合を生み出す極めて近傍以外働かない決して強くはない力です. しかし, 接触の場合のように接触面を作る分子が極めて近くにいる場合, 小さくはない力を発揮します. 分子間力は電気的に中性な分子が電荷の分布の揺らぎによって発生した電気双極子間の相互作用

の揺らぎの平均値により計算されます.

このように, 摩擦の法則は説明されていますが, 接触面が濡れている場合など, その状態によって複雑な振る舞いをするため, 現在でも精力的に研究されています.

少し古い本ですが, 河野彰夫：摩擦の科学（裳華房）が参考に成ります.

## 講評

答案にはネット等で調べたであろう, 言葉（用語）がばちりばめられた記述が少なからず見られました. 残念ながら, 単に, 調べたものをそのまま書き写しているような答案もあり, 調べたものをしっかりと理解し, 自分のものとして自分の言葉で書いていないものは低い評価となっています. 一方, 法則を検証するために, 様々な面積, 荷重, 表面の粗さを変えてシステマティックな実験を考案し, 実行している答案もあり, 正しい法則が得られている答案もありました. このような答案は高く評価しました.

## 6　課題 4A

(会場課題)

会場に物体浮揚実験キットが数台用意してあります. 実際に物体を浮揚させ, 以下の点について物理的に考察してください. .

**問 1** どのような原理で物体は浮揚しているのか, 物体が浮揚している位置について説明しなさい.

**問 2** 物体が浮揚する条件は何でしょうか?

**問 3** 液体を浮揚させた場合を考えます. 液滴の大きさによって形はどのように変わるのか. また, その形はどのように決まっているか.　を考察してください.

## 解説

会場に用意した物体浮揚実験キットは Asier Marzo, Adrian Barnes and Bruce W. Drinkwater : “TinyLev: A multi-emitter single-axis acoustic levitator”, REVIEW OF SCIENTIFIC INSTRUMENTS 88, 085105 (2017). （https://doi.org/10.1063/1.4989995 から無料でダウンロードできます）が元になっています．この論文の中で紹介されている実験キットは amazon などネット通販で販売されています．ただし, 組立説明書は付属していないので youtube などの組み立て解説動画（英語）等を参照する必要があります．この装置は最も小さなサイズで, より大きな筐体の CAD データはネットからダウンロードできるので 3D プリンタで作成できます．

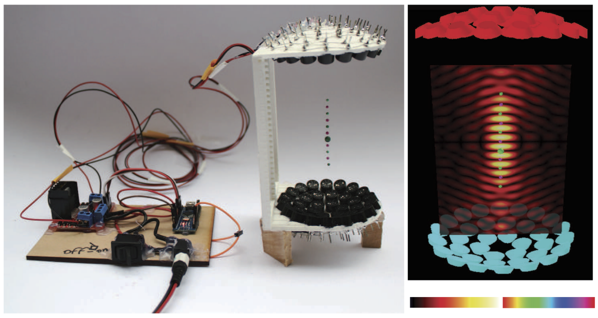


図 1: (a) 制御装置と上下それぞれ 36 個のトランスデューサーからなる単軸浮上装置．発泡ポリスチレン粒子はノードに捕捉される．(b) 数値計算された音場．上下の各円は直径 10mm のトランスデューサーを表し, 色はトランスデューサーの放射位相を表す．（上記文献 FIG.1 より）

図 1 (b) のように, 装置の中央部分には音の定在波が立っています．音場とは圧力の変化なので, 物体をその場所に置くとその定在波の圧力が物体の表面に垂直にかかります．表面に垂直にかかる力を全ての表面について足し合わせる（実際には積分する）と物体が音場から受ける力が求められます（実際には物体を置くことにより音場が乱されるのでその効果を考慮する必要があります）．水の中にある浮沈子が水から受ける力と似ています．

物体の表面に垂直な力を足し合わせたときに横方向の力が残れば横に弾き飛ばされます．また, 横方向の力が残らない状態で, 足し合わせた縦（鉛直）方向の力が重力より大きければ物体は上昇し, 上昇したときに上向きの力が小さくなるならばどこかで釣り合います．逆に, 重力より小さければ落ち, 物体の位置が下がった時に力が大きくなれば重力と釣り合います．以上のことを考えると, 物体が静止する位置は装置の中央付近で音場（圧力）が高い部分と低い部分の間に捕捉されると考えられます．

大気中に置いた, 高さの違いを考慮しなくていいという意味で小さな球には表面に垂直に同じ圧力がかかると考えていいでしょう．その場合の力の合計は大気圧に表面積をかければ得られます．このように, 物体が受ける力には表面積が重要に成ります．いくら軽くても表面積が小さければ浮きません．例えば, 2cm のシャープペンシルの芯は浮きません．しかし, 小さな発泡スチロールのかけらにそのシャープペンシルの芯を通せば中に浮かせることができます．これは発泡スチロールが表面積を持っていたからです．ただ, 力を合計するための表面積なのでフラクタルのような複雑な形にして表面積を大きくしても意味がありません．球形の表面半分に垂直にいたるところ同じ力がかかる場合（何

故半分かというと全体であれば向きが逆になるから力の合計は0になるから, 出ないと球は移動してしまう）, 半球にかかる力の合計は表面に垂直な力に中心を通る断面の面積をかけたものに成ります. 要するに, 求める力の方向から光を当てたときにできる陰の面積に成ります.

密度 $\rho$ が一定, 半径 $r$ の球を考えて, 浮力が射影面積に比例するとすれば, 力の釣り合いは大雑把に

$$\text{重力} = mg = \frac{4}{3}\pi r^3 \rho g = \text{音場の浮力} = \pi r^2 \Delta p$$

と書けます. ただし, $\Delta p$ は上下の圧力差に関係する比例定数と思ってください. これから,

$$\rho r = \text{一定}$$

と成ります. 浮くかどうかは質量だけで決まるわけでも, 密度だけで決まるわけでもないようです.

捕捉させる物体の形状が複雑な場合, より難しく成ります.

## 講評

当日はいろいろ楽しんでもらえたようです. 答案の中には, 非常にシステマティックに実験を行った例もありました（これだけのことをやるにはかなり独占して使っていたのだろうな... と懸念しつつ）. この問題では音が圧力変化で, その圧力が物体を押し上げているということを認識できているかどうかがポイントに成ります. この点に気付いていた答案は少なくありませんでした. そこから, 浮上力には表面積が関係していて, 面積が大きいことが必要だということに気付けるかどうか, 答案に記述しているかどうかが評価ポイントに成ります. この点をしっかり書けている答案もいくつかありました. また, 浮上させる物体の密度に注目した答案は表面積に気付いた答案よりも多くありました. これら3つの点について総合的によくまとまった答案を高く評価しました.

# 7　課題 4B

(遠隔課題)

与えられた問題を計算機で実行する算法の手順 (アルゴリズム) の実行に要する基本処理の手数の総数によって問題の分類を行うことが行われます. 整数の演算を例にアルゴリズムの手数を見積もる手法を考えましょう. 10 進数の乗算を考えます. この時, $1 \times 1$ から始まり $9 \times 9$ で終わる 1 桁の整数の乗算が基本に成ります.

10 進数

$$X = x_{n-1} \times 10^{n-1} + x_{n-2} \times 10^{n-2} + \cdots + x_1 \times 10 + x_0$$

を $X = \langle x_{n-1}, x_{n-2}, \cdots, x_1, x_0 \rangle$ と表すことにします. ただし, $0 \leq x_i < 9 (i = 0, 1, 2 \cdots, n-2)$ $x_{n-1} \neq 0$ です.

**問 1** 2 数 $X = \langle x_{n-1}, x_{n-2}, \cdots, x_1, x_0 \rangle$ $Y = \langle y_{n-1}, y_{n-2}, \cdots, y_1, y_0 \rangle$ の積を計算するのに要する積の回数 (九九の表を参照する階数) を求めなさい. ただし, 計算では桁上がりを考えません.

$0 \leq x_i, y_i < 10$, $n \geq 1$ に対しては, 2 桁の零以上の整数は

$$x = x_1 \times 10 + x_0, \;\; y = y_1 \times 10 + y_0 \tag{1}$$

と表現されます. 従って, 2 数の桁上がりを考えない積は

$$x \times y = x_1 \times y_1 \times 10^2 + (x_1 \times y_0 + x_0 \times y_1) \times 10 + x_0 \times y_0 \tag{2}$$

となり, 積の回数は $\sharp\times = 4$ です.

積

$$\begin{aligned} Z &= z_2 \times 10^2 + z_1 + z_0 \\ Z &= X \times Y \end{aligned}$$

に対して

$$\begin{aligned} z_2 &= x_1 \times x_2 \\ z_1 &= x_0 y_1 + x_1 y_0 \\ &= (x_0 + x_1) \times (y_0 + y_1) \\ &= x_0 y_0 + x_0 y_1 - x_1 y_0 + x_1 y_1 \\ &= z_0 + z_1 + z_2 \\ z_0 &= x_0 \times y_0 \end{aligned}$$

より,

$$\begin{aligned} z_2 &= x_1 \times y_1 \\ z_1 &= (x_1 + x_0) \times (y_1 + y_0) + (-z_2) + (-z_0) \\ z_0 &= x_0 \times y_0 \end{aligned}$$

であるから, 工夫をすれば積の回数を 3 回に削減できることが分かります.

数を表現する基数を $D$ と記すことにします. 十進法では $D=10$, 二進法では $D=2$ であり, インターネットのアドレスでは $D=256=2^8$ が利用されます.

4 桁の数 $\langle x_3,x_2,x_1,x_0\rangle$ は

$$X=x_3D^3+x_2D^2+x_1D+x_0$$

であるから

$$\begin{aligned}X &= x_{21}D^2+x_{20}\\ x_{21} &= x_3D+x_2\\ x_{20} &= x_1D+x_0\end{aligned}$$

と表現できます. 2 数 $X=x_{21}D^2+x_{20}$ $Y=y_{21}D^2+y_{20}$ の積 $Z=z_{22}D^4+z_{21}D^2+z_{20}$ は

$$\begin{aligned}z_{22} &= x_{21}\times y_{21}\\ z_{21} &= (x_{21}+x_{20})\times(y_{21}+y_{20})+(-z_{22})+(-z_{20})\\ z_{20} &= x_{20}\times y_{20}\end{aligned}$$

によって, 2 桁の数の積 3 回で計算できます. そして, 3 桁の数は 3 回の 3 桁の数の積 3 回で計算できます. $n=2^m$ 桁の数の積を計算するために九九を参照する階数を $T(n)$ とすれば,

$$\begin{aligned}T(1) &= 1,\\ T(2) &= 3,\\ T(4) &= T(2^2))=3T(2)=3T\left(\frac{4}{2}\right)\end{aligned}$$

が成立していることが分かります.

**問 2** 基数 $D$ より $D^n$ を利用して, 大きな数を 2 分割して計算を実現することを行うと, $n$ 桁の数同士の乗算に要する時間 $T(n)$ と $2n$ 桁の数同士の乗算に要する時間 $T(2n)$ との間には,

$$T(2n)=3T(n)+cn,\ \ T(1)=1$$

$n=2^m$ が成立することを説明しなさい. ただし, $c$ は定数であり, $cn$ は計算の中間で得た値の加算を実行し並べ替える手間が, 数の桁数 $n$ に依存することを表しています.

**問 3** $c$ を零として, すなわち, 加算と数の並べ替えは瞬時に実行されるとした場合 $T(n)$ を $n$ の関数として

$$T(n)=Cn^\alpha+n\text{ の }\alpha\text{ 次以下の多項式部分}$$

と表すことを考えます. $\alpha$ は 2 以下であることを説明しなさい.

**問 4**(高校生を対象とした問) $n=2^m$ とする. $n$ 桁の数を上位 $n/2$ 桁と下位 $n/2$ に分けて計算することを考える. $c=0$ とすれば,

$$T(2n)=3T(n)$$

より, $T(n)=An^{\log_2 3}$ になることを示しなさい. ただし, $A$ は定数です.

次に, 3 桁の零以上の整数

$$X = x_2 \times 10^2 + x_1 \times 10 + x_0,\ Y = y_2 \times 10^2 + y_1 \times 10 + y_0$$

の桁上がりを考えない積

$$\begin{aligned} Z &= z_4 \times 10^4 + z_3 \times 10^3 + z_2 \times 10^2 + z_1 + z_0 \\ Z &= X \times Y \end{aligned}$$

を考えましょう. 基数 10 を $D$ と表せば, $X$, $Y$ は共に, $D$ の 2 次多項式, $Z$ は 4 次多項式と考えることができるので,

$$\begin{aligned} X(D) &= x_2 D^2 + x_1 D + x_0 \\ Y(D) &= y_2 D^2 + y_1 D + x_0 \\ Z(D) &= z_4 D^4 + z_3 D^3 + z_2 D2 + z_1 D + z_0 \end{aligned}$$

と置けば,

$$Z(D) = X(D)Y(D)$$

です. ここで, $X(D)$, $y(D)$ の係数から $Z(D)$ の係数を求めることが, 2 数の積を計算することに成ります. $Z(D)$ に $D = -2, -1, 0, 1, 2$ を代入すれば, 連立方程式

$$\begin{aligned} Z(-2) &= 16z_4 - 8z_3 + 4z_2 - 2z_1 + z_0 \\ Z(-1) &= z_4 - z_3 + z_2 - z_1 + z_0 \\ Z(0) &= 0z_4 - 0z_3 + 0z_2 - 0z_1 + z_0 \\ Z(1) &= z_4 + z_3 + z_2 + z_1 + z_0 \\ Z(2) &= 16z_4 + 8z_3 + 4z_2 + 2z_1 + z_0 \end{aligned}$$

を得ます. 証明は省きますが, この連立方程式は解を持ち,

$$\begin{aligned} z_4 &= (Z(-2) - 4Z(-1) + 6Z(0) - 4Z(1) + Z(2))/24 \\ z_3 &= (-Z(-2) + 2Z(-1) + 0Z(0) - 2Z(1) + Z(2))/12 \\ z_2 &= (-Z(-2) + 16Z(-1) - 30Z(0) + 16Z(1) - Z(2))/24 \\ z_1 &= (Z(-2) - 8Z(-1) + 0Z(0) + 8Z(1) - Z(2))/12 \\ z_0 &= 0Z(-2) + 0Z(-1) + Z(0) + 0Z(1) + 0Z(2) \end{aligned}$$

です. そして,

$$\begin{aligned} Z(-2) &= X(-2)Y(-2) = (4x_2 - 2x_1 + x_0) \times (4y_2 - 2y_1 + y_0) \\ Z(-1) &= X(-1)Y(-1) = (x_2 - x_1 + x_0) \times (y_2 - y_1 + y_0) \\ Z(0) &= X(0)Y(0) = x_0 \times y_0 \\ Z(1) &= X(1)Y(1) = (x_2 + x_1 + x_0) \times (y_2 + y_1 + y_0) \\ Z(2) &= X(2)Y(2) = (4x_2 + 2x_1 + x_0) \times (4y_2 + 2y_1 + y_0) \end{aligned}$$

を考えれば, 積の数が 5 回で済むことがわかります.

**問 5** $n = 2^m$ 桁の数を数を 2 分割して積を計算する場合に習って, $n = 3^m$ 桁の数を 3 分割して計算をする場合積の回数について

$$T(n) = 5T(n/3)$$

が成立することを説明しなさい.

3 桁の 2 つの零以上の 2 数の積を計算した手法に習って, さらに桁数の多い数の乗算の積の回数を解析しましょう. まず, 5 桁の 2 数 $X = \langle x_4, x_3, x_2, x_1, x_0 \rangle$, $Y = \langle y_4, y_3, y_2, y_1, y_0 \rangle$ とそれらの積 $Z = \langle z_{10}, z_9, z_8, z_7, z_6, z_5, z_4, z_3, z_2, z_1, z_0 \rangle$ を $D$ の多項式で表すと,

$$\begin{aligned} X(D) &= x_4D^4 + x_3D^3 + x_2D^2 + x_1D + x_0 \\ Y(D) &= y_4D^4 + y_3D^3 + y_2D^2 + y_1Dy_0 \\ Z(D) &= z_{10}D^{10} + z_9D^9 + z_8D^8 + z_7D^7 + z_6D^6 + z_5D^5 + z_4D^4 + z_3D^3 + z_2D^2 + z_1D + z_0 \end{aligned}$$

となる. $D$ に $0, \pm 1, \pm 2, \mathbf{3}, \pm 4, \mathbf{5}$ を代入すると $\{z_i\}_{i=0}^{10}$ を未知数とする 11 元連立方程式を

$$z_{10}d_i^{10} + z_9d_i^9 + z_8x_i^8 + z_7d_i^7 + z_6d_i^6 + z_5d_i^5 + z_4d_i^4 + z_3d_i^3 + z_2d_i^2 + z_1d_i + z_0 = Z(d_i)$$

$i = 0, 1, 2, 3, \cdots, 10$ ただし,

$$d_0 = -5, d_1 = -4, d_2 = -3, d_3 = -2, d_4 = -1, d_5 = 0, d_6 = 1, d_7 = 2, d_8 = 3, d_9 = 4, d_{10} = 5$$

とする. このとき, 詳細は省略するが, この連立方程式の解を書き下すことができて,

$$z_i = l_{i10}Z(d_{10}) + l_{i9}Z(d_9) + l_{i8}Z(d_8) + l_{i7}Z(d_7) + l_{i6}Z(d_6) + l_{i5}Z(d_5) + l_{i4}Z(d_3) + l_{i2}Z(d_2) + l_{i1}Z(d_1) + l_{i0}Z(d_0)$$

となる. ここで, $Z(d_i) = X(d_i)Y(d_i)$ である.

**問 6**

2 つの 5 桁の零以上の整数の積を上にまとめた方法で計算する場合, 積の回数は何回必要か見積もってください. ただし, $\{Z(d_j)\}_{j=0}^{10}$ から $z_i$ を計算する重み付和の計算において, 重みをかける部分の積は考えないことにします.

**問 7**

$p$ を素数とする. 2 つの $p$ 桁の零以上の整数の積を計算する場合, 積の回数は何回必要かを見積もってください.

2 次関数

$$y = a_2x^2 + a_1x + a_0$$

が 3 点 $(-1, y_0)$, $(0, y_1)$, $(1, y_2)$ を通過するとき, $a_0$, $a_1$, $a_2$ を決めて関数を決定することを考えよう. 2 次関数の係数が決まれば, $x_0$, $x_1$, $x_2$ と異なる点 $x_4$ で値を計算できます. さて, 通過する点の値を 2

次関数に代入すれば, 3 元の連立方程式

$$\begin{aligned} a_0 - a_1 + a_2 &= y_0 \\ a_0 &= y_1 \\ a_0 + a_1 + a_2 &= y_2 \end{aligned}$$

を得ます. この方程式の解は

$$\begin{aligned} a_0 &= y_1 \\ a_1 &= -\frac{1}{2}y_0 + \frac{1}{2}y_2 \\ a_2 &= \frac{1}{2}y_0 - y_1 + \frac{1}{2}y_2 \end{aligned}$$

です.

2 桁の零以上の 2 数の乗算は, まず数 $X = \langle x_1, x_0 \rangle$, $Y = \langle y_1, y_0 \rangle$, を多項式

$$X(d) = x_1 d + x_0 \quad (3)$$

$$Y(d) = y_1 d + y_0 \quad (4)$$

と考えて, 3 点 $(-1, X(-1)Y(-1))$, $(0, X(0)Y(0))$, $(1, X(1)Y(1))$, 次いで, を通過する 2 次関数

$$Z(d) = X(d)Y(d) = z_2 d^2 + z_1 d + z_0$$

の係数 $z_2$, $z_1$, $z_0$ を決めることに成ります. そして同様に, 3 桁の零以上の 2 数の乗算は数 $X = \langle x_2, x_1, x_0 \rangle$, $Y = \langle y_2, y_1, y_0 \rangle$, を多項式として 5 点 $(-2, X(-2)Y(-2))$, $(-1, X(-1)Y(-1))$, $(0, X(0)Y(0))$, $(1, X(1)Y(1))$, $(2, X(2)Y(2))$, を通過する 4 次関数

$$Z(d) = z_4 d^4 + z_3 d^3 + z_2 d^2 + z_1 d + z_0$$

の係数を決めることに成ります.

## 解説

自然数の乗算の基本は $1 \times 1$ から $9 \times 9$ までの表 (九九表) の参照と桁上がりの加算です. さて, 2 桁の 2 数

$$X = x_1 \times 10 + x_0, \ \ Y = y_1 \times 10 + y_0$$

ただし, $1 \le x_1, x_2 \le 9$, $1 \le y_1, y_2 \le 9$ の乗算は,

$$\begin{aligned} X &= x_1 \times 10 + x_0 \\ Y &= y_1 \times 10 + y_0 \\ Z &= z_2 \times 10^2 + z_1 \times 10 + z_0 \end{aligned}$$

と置けば

$$\begin{aligned} z_2 &= x_1 \times y_2 \\ z_0 &= x_0 \times y_0 \\ z_1 &= x_1 \times y_0 + x_0 \times y_1 \end{aligned}$$

より, 乗算が 4 個含まれるため, 九九表を 4 回参照します. また, 4 回以上は必要ありません. これは, 2 つの自然数の積を計算する場合, 桁数の大きい方の数の桁数を $n$ とすれば, 最大 $n^2$ 回の九九表参照が必要であることを表しています.

しかし, 工夫をすれば

$$\begin{aligned} z_2 &= x_1 \times y_2 \\ z_0 &= x_0 \times y_0 \\ z_1 &= z_2 + z_0 + (x_1 - x_0) \times (y_1 - y_0) \end{aligned}$$

となり, 九九表を 3 回参照するだけで済みます. この方法を Karatsuba の方法といいます.

$n = 2^m$ のとき,

$$\begin{aligned} A &= a_{n-1}10^{n-1} + a_{n-2}10^{n-2} + \cdots + a_1 10^1 + a_0 \\ &= (a_{n-1}10^{(n-1)/2} + \cdots + a_{((n-1)/2)})(a_{(n-1)/2-1}10^{(n-1)/2-1} + \cdots + a_0) \end{aligned}$$

とすれば, 上の算法を繰返し適用すると, $n$ 桁の自然数同士の乗算を計算するための九九表参照の回数 $M(n)$ と加算の演算回数 $A(n)$ に対して, 漸化式

$$\begin{aligned} M(n) &= 3M\left(\frac{n}{2}\right), \ M(1) = 1 \\ A(n) &= 3A\left(\frac{n}{2}\right) + 8n, \ A(1) = 0 \end{aligned}$$

が成立します. この漸化式の一般項は

$$\begin{aligned} M(n) &= n^{\log_2 3} \\ A(n) &= 8(n^{\log_2 3} - n) \end{aligned}$$

です．$M(n)$ の計算には，$M(2^m) = f(m)$ に対して，$f(m) = 3f(m-1)$ と置けば，$f(m) = 3^m f(1)$ より，$M(n) = 3^{\log_2 n}$ を得ます．次いで，$a^{\log_b n} = n^{\log_b a}$ を利用すれば，$M(n) = n^{\log_2 3}$ が導けます．$M(n)$ を $n$ の実数次の冪関数として表現できます．通常の計算で九九表の参照回数と加算の処理回数は，

$$
\begin{aligned}
M(n) &= n^2 \\
A(n) &= 2n(n-1)
\end{aligned}
$$

です．$n^{\log_2 3} < n^{\log_2 2^2} = n^2$ より，$n$ 桁の 2 数の積を計算する乗算の回数は，工夫をすれば $n^2$ 回より少なくなることが分かります．

十進数を表す

$$
A = a_{n-1}10^{n-1} + a_{n-2}10^{n-2} + \cdots + a_1 10 + a_0 10^0
$$

の 10 を基数といいます．基数を $d$ で表せば，2 数

$$
\begin{aligned}
A(d) &= a_{n-1}d^{n-1} + a_{n-2}d^{n-2} + \cdots + a_1 d + a_0 \\
B(d) &= b_{n-1}d^{n} + b_{n-2}d^{n-2} + \cdots + b_1 d + b_0
\end{aligned}
$$

の乗算を多項式の積

$$
\begin{aligned}
C(d) &= A(d)B(d) \\
&= c_{2n-1}d^{2n-1} + \cdots + c_1 d + c_0
\end{aligned}
$$

として表すことができます．このとき，

$$
c_k = \sum_{i=0}^{k} a_{k-i}b_i = \sum_{i=0}^{k} b_{k-i}a_i
$$

です．

$m$ 次多項式は $m+1$ 個の $d$ に於ける値が分かれば，その係数がすべて求まります．値を与える点を $\{d_i\}_{i=0}^{2n-1}$ とすれば，線形方程式

$$
c_{2n-1}d_i^{2n-1} + \cdots + c_1 d_i + c_0 = A(d_i)B(d_i), \;\; i = 0, 1, \cdots, 2n-1
$$

を解けば $C(d)$ の係数が求まり，$A(d)B(d)$ の積が求まります．実際，2 つの 2 桁の積は $\{d_i\}_{i=0}^{2} = \{-1, 0, 1\}$ とすれば，

$$
\begin{aligned}
c_0 &= a_0 b_0 \\
c_2 - c_1 + c_0 &= (a_1 + a_0)(b_1 + b_0) \\
c_2 + c_1 + c_0 &= (a_1 - a_0)(b_1 - b_0)
\end{aligned}
$$

より，

$$
\begin{aligned}
c_0 &= a_0 b_0 \\
c_1 &= a_1 b_1 + a_0 b_0 - (u_1 - u_0)(v_1 - v_0) \\
c_2 &= a_1 b_1
\end{aligned}
$$

として求まります. Karatsuba の方法を導くことができます.

Karatsuba の方法に習って 2 つの 3 桁の数の積を計算する場合, 4 次方程式の係数を計算することに成ります. 4 次方程式の 5 点での値を計算するために 2 次方程式の積を 5 回計算す必要があります. 従って, 整数を 3 分割して積を計算する場合, 乗算に関する計算量に関する方程式は

$$M(n) = 5M\left(\frac{n}{3}\right)$$

となり,

$$M(n) = Cn^{\log_3 5}$$

を得ます. そして,

$$n^{log_3 5} \approx n^{1.465} < n^2$$

が成立します.

線形方程式を解く方法を考察しましょう. 降冪

$$x^{\underline{n}} = x(x-1)(x-2)\cdots(x-(n-1)),\ \ x^{\underline{0}} = 1$$

を定義します. そして, 差分を

$$\Delta f(x) = f(x+1) - f(x)$$

とすれば, 降冪に対して,

$$\Delta x^{\underline{n}} = nx^{\underline{n-1}}$$

が成立します.

第 2 種 Stirling 数を用いれば, 降冪と冪との間に

$$x^n = \sum_{k=0}^{n} \left\{ \begin{array}{c} n \\ k \end{array} \right\} x^{\underline{k}}$$

が成立することが知られています. ただし, 第 2 種 Stirling 数は

$$\left\{ \begin{array}{c} n \\ k \end{array} \right\} = \left\{ \begin{array}{c} n-1 \\ k-1 \end{array} \right\} + k \left\{ \begin{array}{c} n-1 \\ k \end{array} \right\},\ \ \left\{ \begin{array}{c} 0 \\ 0 \end{array} \right\} = 1$$

によって定義されます.

まず,

$$C(x) = \sum_{k=0}^{2n-1} c_k x^k = \sum_{k=0}^{2n-1} w_k x^{\underline{k}}$$

に対して,

$$\Delta^k c(0) = k! w_k$$

が成立します. そこで,

1. $C(0)$ の値を 0 行目に並べる.

2. $(k-1)$ 列目の隣合う要素の差を $k$ で割った数を $k$ 行目に並べる.

3. $k$ 行目の左端の数が $w_k$ となる.

4.

$$
\begin{aligned}
& w_n x^{\underline{n}} + w_{n-1} x^{\underline{n-1}} + \cdots + w_0 \\
= \quad & (\cdots(w_n(x-n) + w_{n-1}(x-(n-1)) + w_{n-2}(x-(n-2)) \cdots w_n) \cdots) x + w_0 \\
= \quad & c_n x^n + c_{n-1} x^{n-1} + \cdots + c_1 x + c_0
\end{aligned}
$$

を計算すれば, 多項式の係数を計算できます.

以上の演算は $\{d_i\}_{i=0}^{2n} = \{0, 1, 2, \cdots 2n\}$ と置いて, 逆行列を計算して, 多項式の係数を計算したことに相当しています.

整数の乗算は算術計算の基本となる演算です. 基本的な演算の手間をデータ量 (乗算の場合は桁数) によって評価し, ある問題と他の問題との間の手間の複雑さを評価できることに成ります. 一見, アルゴリズムを構成できそうもない問題でも, 手間を評価して同じ手間でアルゴリズムを実現可能な問題があれば, 考えている問題のアルゴリズムが構成できることがあります.

乗算の高速化は, 上に述べたように高等学校数学を巧妙に組み立てることで実現できます. このほかにも, $n \times n$ 次の正方行列の乗算ための積の演算回数 $T(n)$ は

$$\alpha n^2 < T(n) \leq \beta n^3$$

を満たすため

$$T(n) = \gamma n^{2+\epsilon}$$

の $\epsilon$ をどこまで 0 に近づけることができるのか研究が続いています.

問題をデータ数を基準にして規模の小さな問題に分割して計算する場合の計算複雑性の計算法として Master 方程式を使った計算法が一般的です. データ数に依存したある問題の計算時間 (乗算の回数を基本にすることが多い. ) を $T(n)$ とします. データを $b$ 分割してから, 各分割に同じ処理を繰り返す回数を $a$ とします. さらに, $b$ 分割した解けた問題から, 元の問題の解を構成する手間を $f(n)$ とすれば,

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

が成立します. $f(n)$ は $n$ に関して多項式で評価できる場合がほとんどです. そこで

$$T(n) = aT\left(\frac{n}{b}\right) + n^c$$

と置いて, $T(n)$ を求めてみます. $r = a/b^c$ と置くと,

$$
\begin{aligned}
T(n) \quad & = \quad n^c \sum_{k=1}^{\log_b n} r^k \\
& = \quad \begin{cases} \Theta(n^c), & r < 1, \quad c > \log_b a \\ \Theta(n^c \log n), & r = 1, \quad c = \log_b a \\ \Theta(n^{\log_b a}), & r > 1, \quad c < \log_b a \end{cases}
\end{aligned}
$$

を得ます．この解は

$$
\begin{aligned}
\sum_{k=0}^{n} r^k &\leq \frac{1}{1-r},\ \ r<1 \\
\sum_{k=0}^{n} r^k &= k+1,\ \ r=1 \\
\sum_{k=0}^{n} r^k &= \frac{r^{k+1}-1}{r-1},\ \ r>1
\end{aligned}
$$

から導かれます．また $f(n) = \Theta(g(n))$ とは, ある数 $n_0$ より大きい全ての $n$ に対して, 定数 $c_1$, $c_2$ が存在して, $c_1 \times g(n) \leq f(n) < c_2 \times g(n)$ が成立することです．このことは, $n$ が大きくなれば, $f(n)$ の挙動が上と下から $g(n)$ の定数倍で挟まれることを意味します．すなわち, $n$ が大きくなった場合の $f(n)$ の挙動が $g(n)$ によって決まること意味しています．例えば, $g(n) = n$ であれば, $f(n)$ は $n$ が大きくなった場合に $f(n) = An + B$ となり, 線形 (一次) 関数になることを意味しています．

物理学や工学で重要な情報となる, 時間関数や振動の周波数による解析のために利用されるフーリエ変換を数値的に高速に計算するための高速フーリエ変換の Master 方程式は

$$T(n) = 2T\left(\frac{n}{2}\right) + cn$$

と成ります．さて,

$$T(n) = 2T\left(\frac{n}{2}\right) + cn$$

を解いてみましょう．

$$\frac{T(n)}{n} = \frac{T(n/2)}{n/2} + c$$

を得るので,

$$\frac{T(2^k)}{2^k} = T(1) + ck$$

と成ります．さらに変形を進めると,

$$T(2^k) = n(ck+t),\ \ t = T(1),\ \ k = \log n$$

となります $2^k = n$ より, $k = \log n$ を代入すれば,

$$T(n) = \Theta(n \log n)$$

を得ます．計算複雑性のオーダー ($n$ に関する最大次数) が $\Theta(n \log n)$ となります．一般に, 問題を半分に分けて同じ処理を行い, 結合して元の問題を解く計算のオーダーは $\Theta(n \log n)$ に成ります．

また,

$$
\begin{aligned}
T(n) &\leq 2T(n/2) + cn \\
&\leq 2(2T(n/2^2) + c(n/2)) + cn = 2^2T(n/2^2) + 2cn \\
&\leq 2^2(2T(n/2^3) + c(n/2^2)) + 2cn = 2^3T(n/2^3) + 3cn \\
&\leq \cdots
\end{aligned}
$$

$$
\begin{aligned}
&\leq 2^k T(n/2^k) + kcn \\
&= 2^k T(1) + ckn \\
&= dn + cn \log_2 n \\
&= O(n \log_2)
\end{aligned}
$$

として, $n$ に関する不等式を得ます．ここで，$2^k = n$, $(k = \log_2 n)$, $T(1) = d$ です．ただし, $f(n) = O(g(n))$ とは, $n \geq n_0$ に対して、定数 $\alpha > 0$ が存在して $|f(n)| \leq \alpha O(g(n))$ が成立することです.

Karatsuba の問題を拡張して, 多数桁の整数を 3 分割して乗算を計算する問題は, 次数の低い多項式に値を代入して, 5 回乗算を実行することになるので, 乗算に関する Master 方程式は

$$
M(n) = 5M\left(\frac{n}{3}\right)
$$

に成ります．さらに, 多数桁の整数を $p$ 分割して多項式の補間を行う計算法の積の回数に関する Master 方程式は

$$
M(n) = (2p-1)M\left(\frac{n}{p}\right)
$$

となり, $M(n) = n^{\log_p(2p-1)}$ に成ります.

## 講評

難しかったようです.

# ロボットの部

## 8　ロボットの部

第 1 日目の実習に続いて, 第 2 日目に後に以下の課題を課しました. 第 2 日目 15:00 から競技を開始しました.

**課題 1** 1.5m の直線走路をできるだけ早く走行できるように, プログラムしてください.　但し, テープで示されたレーンから Rapiro の両足がはみ出ると失格と成ります.

**課題 2** 床にテープで描いた迷路を通ってできるだけ短時間でゴールに到達するようにプログラムしてください. 但し, テープで示されたレーンから Rapiro の両足がはみ出ると失格と成ります.

**課題 3**　Rapiro をプログラムしてダンスや体操などのパフォーマンスをさせてください. 但し, Rapiro を制御する技術面と, パフォーマンスの印象面を総合して判定します.

## 解説

COVID-19 の影響で, ロボットの部はしばらく開催されていませんでしたが, 今年やっと復活しました. 初心者の人にはプログラムでロボット Rapiro を制御することを体験してもらい, 機器を制御する面白さを体験してもらうことを, プログラムに詳しい人には Rapiro がより高いパフォーマンスを発揮できるように様々に工夫してもらうことを狙いに開催しています.

## 講評

１つ目の種目は, 1.5 m走でした. レーンからはみ出さずにできるだけ早く 1.5 m先のゴールにたどり着くことを競いました. 床面が完全な水平ではなかったので真っすぐ前進すること自体が簡単ではなかったようです. 今回は, 通常の直立歩行だけではなく, 匍匐前進をするロボットも登場して, タイムを競いました. トーナメント形式で行い１位を決定しました.

２つ目の競技は, 床面にテープを貼ってつくった迷路をできるだけ早く抜けるものです. Rapiro に内蔵の距離センサーでテープを認識しながら迷路を進むパターンと, あらかじめ迷路の形状を織り込んでプログラム通りに走り抜けるパターンなどのトライがありました. 残念ながら, ゴールまでたどりつけたロボットはありませんでしたが, 一番長い距離を進めたものを１位としました. （やはり床面の傾きが災いしたようです）

３つ目の競技は, ダンス競技で, 自由にダンス・体操などを演技してもらい, ロボットの部スタッフと参加者ならびに見学者の投票により最も点数が高かった Rapiro を１位としました.

今回は, ロボット Rapiro を如何にうまく制御できるかを, 3 つの課題を通して評価し, 各課題につき１名ずつ機巧賞としました. また, 競技直前に Rapiro が故障して実力を発揮できなかったものの, テスト段階では面白い工夫をして良いタイムを出していたチームがあったので, 敢闘賞を送ることとしました. 来年に向けて, ロボットや競技スタイルなどさらに工夫していきたいと思います.

来年に向けて, ロボットや競技スタイルなどさらに工夫していきたいと思います.

# 人工知能の部

# 9　人工知能の部

**課題 1** 2 つの検査（検査 1，検査 2）結果で風邪と診断されたグループと花粉症と診断されたグループのデータがあります.

**花粉症**

| 個人番号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 検査 1 | 6.2 | 9.9 | 7.7 | 6.9 | 8.9 | 4.3 | 8.6 |
| 検査 2 | 2.3 | 6.7 | 3.2 | 4.5 | 4.3 | 4.6 | 2.0 |

**風邪**

| 個人番号 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|
| 検査１ | 2.1 | 0.9 | 3.1 | 4.1 | 6.2 | 4.9 | 5.1 |
| 検査２ | 6.0 | 5.3 | 3.2 | 5.8 | 1.3 | 3.1 | 1.8 |

1. 花粉症, 風邪を特徴量空間にプロットしなさい.
2. 機械学習を用いて, 横軸に検査 1，縦軸に検査 2 をとり, 花粉症, 風邪の境界線を引きなさい.
3. 機械学習を用いて, 2 つの検査結果が (5.7, 5.0) である場合, どちらと判断されるか, また, それぞれ疾患である確率を求めなさい.
4. 与えられたデータだけからより明確に (5.7, 5.0) の診断を下すにはどのようにすればよいか.
　工夫した（改善された）, 特徴量空間プロット, 花粉症, 風邪の境界線プロット, (5.7, 5.0) の診断確率を示しなさい.

**課題 2** ある病の診断に対して，７つの項目の検査を行い，正常，症状 1，症状 2 と診断されたグループのデータがあります．

**正常**

| 個人番号 | 検査 1 | 検査 2 | 検査 3 | 検査 4 | 検査 5 | 検査 6 | 検査 7 |
|---|---|---|---|---|---|---|---|
| 1 | 33 | 21.21 | 54.55 | 1.5 | 30.30 | 2 | 1.44 |
| 2 | 20 | 35.00 | 25.00 | 0.0 | 45.00 | 6 | 1.95 |
| 3 | 18 | 27.78 | 16.67 | 0.0 | 38.89 | 3 | 2.22 |
| 4 | 31 | 12.90 | 32.26 | 3.5 | 51.61 | 5 | 1.52 |
| 5 | 28 | 10.71 | 50.00 | 0.0 | 46.43 | 7 | 1.23 |
| 6 | 30 | 6.67 | 6.33 | 3.0 | 60.00 | 5 | 1.37 |
| 7 | 27 | 3.70 | 2.22 | 2.0 | 51.85 | 1 | 2.11 |
| 8 | 33 | 6.00 | 4.85 | 2.0 | 57.58 | 2 | 1.32 |
| 9 | 57 | 0.53 | 31.58 | 1.5 | 56.14 | 11 | 1.55 |
| 10 | 20 | 15.00 | 20.00 | 2.0 | 50.00 | 8 | 1.53 |
| 11 | 14 | 14.29 | 28.57 | 4.0 | 57.14 | 4 | 1.71 |
| 12 | 36 | 8.33 | 36.11 | 2.0 | 66.67 | 5 | 1.42 |
| 13 | 30 | 16.67 | 36.67 | 4.0 | 30.00 | 6 | 1.42 |
| 14 | 37 | 0.81 | 45.95 | 4.0 | 54.05 | 7 | 1.41 |
| 15 | 11 | 45.46 | 45.46 | 0.0 | 36.36 | 6 | 2.95 |
| 16 | 17 | 11.77 | 35.29 | 5.0 | 58.82 | 2 | 1.59 |
| 17 | 34 | 8.82 | 32.35 | 1.5 | 55.88 | 6 | 1.34 |
| 18 | 18 | 22.22 | 33.33 | 0.0 | 33.33 | 4 | 1.78 |
| 19 | 27 | 22.22 | 29.63 | 0.0 | 37.04 | 5 | 1.46 |

**症状 1**

| 個人番号 | 検査 1 | 検査 2 | 検査 3 | 検査 4 | 検査 5 | 検査 6 | 検査 7 |
|---|---|---|---|---|---|---|---|
| 20 | 27 | 14.82 | 18.52 | 7.0 | 51.85 | 7 | 1.57 |
| 21 | 47 | 6.83 | 59.57 | 2.0 | 61.70 | 5 | 1.15 |
| 22 | 19 | 10.53 | 5.26 | 2.0 | 68.42 | 4 | 1.16 |
| 23 | 14 | 35.71 | 14.29 | 0.0 | 35.71 | 3 | 1.21 |
| 24 | 11 | 26.67 | 45.45 | 0.0 | 54.55 | 3 | 0.91 |
| 25 | 15 | 19.93 | 33.33 | 1.0 | 60.00 | 7 | 1.13 |
| 26 | 26 | 14.29 | 11.54 | 3.5 | 46.15 | 10 | 1.33 |
| 27 | 7 | 16.00 | 71.43 | 0.0 | 42.86 | 3 | 1.21 |
| 28 | 23 | 16.67 | 65.22 | 0.0 | 43.48 | 4 | 1.04 |
| 29 | 18 | 16.09 | 50.00 | 2.5 | 61.11 | 5 | 1.36 |
| 30 | 11 | 9.09 | 27.27 | 0.5 | 18.18 | 2 | 0.64 |
| 31 | 26 | 11.54 | 26.92 | 6 | 38.46 | 6 | 1.44 |
| 32 | 15 | 13.33 | 53.33 | 1.5 | 40.00 | 4 | 0.83 |
| 33 | 19 | 36.84 | 21.05 | 1.5 | 36.84 | 3 | 1.45 |
| 34 | 19 | 10.53 | 78.95 | 0.0 | 73.68 | 4 | 1.13 |
| 35 | 15 | 6.67 | 53.33 | 0.0 | 46.67 | 5 | 1.23 |
| 36 | 32 | 6.25 | 68.75 | 0.0 | 15.63 | 5 | 1.05 |
| 37 | 33 | 15.15 | 15.15 | 2.0 | 60.61 | 5 | 0.92 |
| 38 | 42 | 28.57 | 23.81 | 1.5 | 35.71 | 5 | 1.48 |

**症状 2**

| 個人番号 | 検査 1 | 検査 2 | 検査 3 | 検査 4 | 検査 5 | 検査 6 | 検査 7 |
|---|---|---|---|---|---|---|---|
| 39 | 11 | 9.09 | 18.19 | 1.5 | 54.55 | 3 | 1.50 |
| 40 | 21 | 0.00 | 36.36 | 3.0 | 59.09 | 4 | 1.27 |
| 41 | 11 | 27.27 | 54.55 | 0.0 | 27.27 | 6 | 1.00 |
| 42 | 13 | 15.39 | 23.08 | 1.0 | 53.85 | 3 | 1.38 |
| 43 | 16 | 6.25 | 43.75 | 1.0 | 93.75 | 4 | 0.94 |
| 44 | 11 | 0.0 | 27.27 | 1.0 | 54.55 | 1 | 1.09 |
| 45 | 13 | 0.0 | 38.46 | 3.0 | 38.46 | 1 | 0.69 |
| 46 | 29 | 3.45 | 41.38 | 2.0 | 51.72 | 5 | 0.50 |
| 47 | 17 | 0.00 | 64.71 | 3.0 | 64.71 | 2 | 0.80 |
| 48 | 22 | 0.00 | 77.27 | 1.0 | 77.27 | 1 | 0.66 |
| 49 | 16 | 6.25 | 56.25 | 0.0 | 25.00 | 1 | 0.28 |
| 50 | 12 | 33.33 | 58.33 | 0.0 | 50.00 | 2 | 1.12 |
| 51 | 18 | 22.22 | 44.44 | 0.0 | 55.56 | 5 | 1.60 |
| 52 | 23 | 0.00 | 39.13 | 1.0 | 47.83 | 4 | 0.87 |
| 53 | 12 | 0.00 | 91.67 | 0.0 | 83.33 | 4 | 0.25 |
| 54 | 7 | 14.29 | 57.14 | 0.0 | 85.71 | 4 | 1.09 |
| 55 | 44 | 13.64 | 59.09 | 0.0 | 40.91 | 1 | 0.41 |
| 56 | 11 | 18.18 | 63.64 | 1.0 | 27.27 | 2 | 0.59 |
| 57 | 24 | 12.50 | 25.00 | 1.0 | 45.83 | 3 | 1.41 |

1. 7 つの項目のうち, どのような組み合わせの何個の項目の検査を組み合わせたデータを用いると, 2 次元もしくは 3 次元の特徴量空間で, 正常, 症状 1, 症状 2 を明確に分類できるか考えなさい.

2. 3 組のデータ

   (24, 33.33, 25.00, 2.5, 41.67, 7, 2.19)
   (12, 0.00, 91.67, 0.0, 75.00, 2, 1.00)
   (24, 12.50, 25.00, 1.0, 45.83, 3, 1.42)

   はそれぞれ, どのような確率でどの状態であると同定されるかを考察してください.

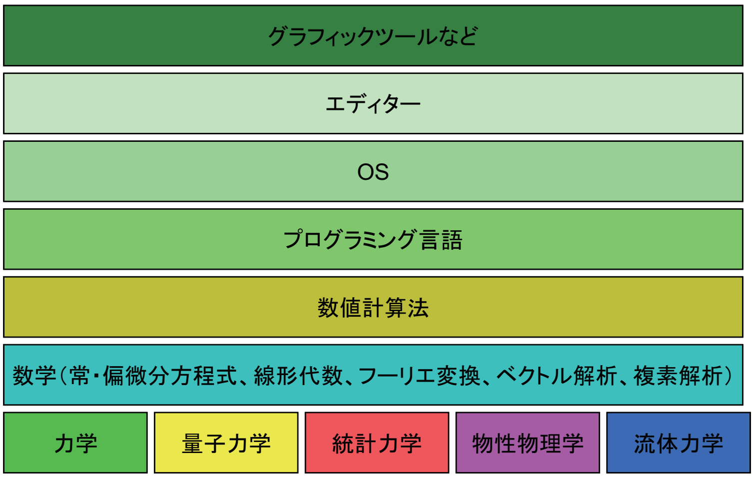
* データは全て「数式処理ソフト Wolfram 言語 a による多変量解析（現代数学社）」より

## 解説

近年, 人工知能の開発が進み, ChatGPT や IBM の Watson などエンドユーザー向けのサービスが始まっています. 好むと好まざるにかかわらず, 望ましいか望ましくないかにかかわらず, そのような時代が始まっています. このコンクールの人工知能の部では人工知能の開発に取り組んで欲しいわけではありません. そうしたとしても, 参加できる中学, 高校生の人数は限られているでしょう. また, 社会においても, AI 開発者の人数よりもエンドユーザーの方が圧倒的に多いでしょう. このコンクールでは後者にどのように使い, どれだけの成果を出せるのかを競ってもらおうという趣旨です.

さて, 自然科学分野でコンピューターを利用するとなるとプログラミングが欠かせません（でした）. コンピューター・シミュレーションを行うとなると, プログラミング以外の様々な知識が必要と成ります. それについて少し説明してみたいと思います.

実際にコンピューター・シミュレーションを実行するために, 何を知っていなければならないでしょうか？コンピューター・シミュレーションを実行する場合, 解析したい現象どのような法則に支配さ




れているのか, 数値計算をするのであれば数式で表す必要があります. モンテ・カルロ法（確率的な方法）を用いるのであれば, 各個体がどのようなときにどのように振舞うのかの行動のルールを決定する必要があります. それには, それぞれ分野に関連した, 力学, 電磁気学, 流体力学, 量子力学, 統計力学, 熱力学, 構造力学, 物性物理学, 原子物理学, 相対論, 化学, 生物学を習得し, 基礎方程式を理解していなければなりません. また, それを解くために必要な数学, 偏微分方程式, 常微分方程式, 線形代数, ベクトル解析, フーリエ変換などを理解しておく必要があります.

その上で, さらに, 数値計算法を勉強しておく必要があります. 数値計算法なるものがなぜ必要かというと, 現在のコンピュータはデジタルコンピュータで, 数値を有限桁の 2 進数とし, 回路の ON, OFF の 2 値で表現し, 計算しており, 数値しか取り扱えません. そもそも, 自然現象を記述する微分方程式の微分も積分もできません. では, コンピュータでどのように微分や積分を表現し, 計算するのか？ただ, 単純な $n$ 次方程式を解くにしても, 数値だけでどのようにすれば方程式が解けるのか方法を考えなければなりません. 電卓で平方根や sin, cos をボタン一つで計算できるが, では, 電卓の中ではどのようにその数値を計算しているのでしょうか？その手順（アルゴリズム）, 方法を考えるのが数値計算法であり, コンピューター・シミュレーションを実行する場合, 必要不可欠な知識です.

コンピューターでの計算結果が信用できるかどうかという問題がよく起こりますが, それはコンピューターの問題ではありません. コンピューターは人間の指示通りの計算を行っています. 計算結果は全て人間の指示によるものであり, 全て実行させた人間の責任です. 基礎的な物理学, 数学の知識がいくら優秀でもこの数値計算法の知識が中途半端な人物が実行したシミュレーションの結果は信用するに値しないと言わざるを得ません. 自分がシミュレーションを実行する場合には, しっかり, 習得してからにすべきであるし, 逆に, 他人の結果を見るときは注意深く確認する必要があります. でなければ, ガセネタをつかまされかねません.

数値計算法もしっかり習得した, では, シミュレーションが実行できるかというとそうはいきません.

コンピュータを操作できなければどうにもなりません. コンピュータが使えるというと, OS (Operating System, 基本ソフト) が使えることを意味するでしょう.

ここまででコンピュータの基本操作はできるが, まだ数値計算はできません. 数値計算をするには, 通常, 連立方程式や微分方程式を解く数値計算法のアルゴリズムをプログラミング言語を用いてプログラムにしなければなりません. 従って, 自分のシミュレーションに最も適したプログラミング言語で自在にプログラムが書けるようにならなければなりません.

プログラミング言語を習得し, 必要なプログラムを書き下せるようになったら, それをコンピューターのファイルに入力しなければなりません. 通常, プログラムはプレーンテキスト（書式なしのテキスト）ファイルで書かれます. 入力用のアプリケーション・ソフト, テキスト・エディターを使って入力します. 通常, 大きなプログラムは機能ごとに複数のファイルに分割して保存します. 何十と言うファイルになってくると, 修正などを行った時に版管理をするのも大変になってくる. そこで, 近年, Windows 上ではプログラミングのための統合環境, Microsoft Visual Studio などが用いられます. このソフトウェアにはエディタ機能も含まれています.

これでプログラムがとりあえず完成しました. しかし, そのままでは動かない. プログラミング言語で書かれたプログラムをコンピュータが実行できるもの（ロードモジュールという）に変換（翻訳）しなければなりません. その機能を持ったソフトウェアをコンパイラ (compiler) といいます. 複数のファイルを各々コンパイラでオブジェクト・モジュールというものに一旦変換し, プログラムに必要なすべてのオブジェクト・モジュールとコンパイラが持っている組み込み関数などのライブラリを合わせて（リンク）, ロードモジュールを作成します. 通常, 初めてコンパイラにかけたとき, いくつもの syntax error（文法の間違い, ミスタイプ）の警告が出るのが普通です. どこが間違えているか, 場所を知らせてくれるので, エディターで修正し, コンパイルするという作業を繰り返し, エラーがなくなれば実行できるプログラム（アプリケーション）が完成します.

これでプログラムを実行すれば正しい解が得られるというものではありません. 通常, アルゴリズムや式の間違いなどで, プログラムが実行途中で異常終了したり, 正常終了しても, 得られた解が正しくないことが少なくありません. 実行が異常終了するときにはどこがおかしいか分かり易いが, 正常終了し, 答がおかしい場合には, どこで間違えているのか見つけるのに困難が伴い, 非常に時間がかかります. この不具合をバグ (bug, 虫) といい, それを解決する作業をデバッグ (debug, 虫取り) と言います. 近年, スマートフォンや自動車, 家電製品の動作が突然おかしくなったりするのは, 中に搭載されているソフトウェアが巨大化, 複雑化し, 完全にデバッグすることが困難（ほぼ不可能）になり, バグが残ったまま発売されているからです.

結果の検証をし, 作成したプログラムが正しい結果を与えるようになれば, 本格的に計算を実行します. 結果は単なる数字の羅列です. 大規模な計算であれば数字の羅列を見てもそれがどのような結果を表しているのかを人が理解することは不可能です. 例えば, 天気予報で各地の緯度, 経度, 高度, 気圧, 気温, 湿度, 風向などの数字の羅列をテレビの画面で流されても何が何だか分かりません. そこで, コンピュータ・シミュレーションでは計算で得られたデータをグラフにしたり, アニメーションにしたり, 可視化する必要があります. そのために, 可視化ツール（アプリケーションソフト）を使えるようになって初めてシミュレーションが完結します.

Wolfram 言語 a はもともと数式処理をおこなうソフトウェアでしたが, ２次元グラフ, 等高線, 密度グラフ, ベクトル図, それらの３次元版などの機能も備えたもので, 微分方程式の数値計算なども式を入力すれば実行してくれるようになっていて, 解く問題設定から以降の知識はなくとも Wolfram 言語 a が関数（コマンド）を知ってさえいれば実行してくれるようになっています. そういう意味では Wolfram 言語 a は自分の解きたい問題に関する知識さえあれば, それ以外の煩雑な作業から解放してくれたと言っても過言ではありません. 微分方程式は書けるけれど, 解く数学お知識がない中高生でも, アイデア次第で一線級の研究活動が可能に成ります. 実際, 外国では Wolfram 言語 a を用いて, 高校生でも話題になるような研究を行っている人は多くいます.

このような機能を持つソフトウェアとしては他にも Matlab, Maple などがありますが, Matlab は工学関係, 企業系で多く利用されています. Matlab も Wolfram 言語同様に人工知能機能を盛んに開発し, 日本でも多くの会社が Matlab の機械学習機能を用いて業務改善に応用しています. 機械学習, 人工知能の知識がなくとも実用的な分類, 診断などに直ぐに使えます. 中高生でもこれらの機能を使って, アイデア次第で世界に通用するシステムを構築することは可能です. このような状況を鑑み, 日本の中高生, 受験勉強ばかりで大丈夫か？という意味で, 一つ, いいアイデアを出して, すごいシステムを作ってみて欲しいと期待しているのです.

世界中の有力な大学, 研究所, 日本でも有力な大学の多くが Wolfram 言語のサイトライセンスを契約し, 学生教員は自由に利用できるようになっています（そのような大学なら, Matlab も契約していると考えられますが...）. 数理生物系の方々の研究のほとんどは Wolfram 言語 a を用いて行われていますし, 物理分野の論文に掲載されている 3 次元グラフの半数, 理工系のテキストに掲載されている３次元グラフの多くが Wolfram 言語を用いて描かれています. Wolfram 言語 a の機械学習機能で分類するのは関数 3 つの知識があれば実行可能です. 後は機能をどう使うかのアイデア次第です. そういう理由で Wolfram 言語を採用しました.

**課題 1**

1 は手はじめに, 関数を知っていますかという問題で, notebook1 のように, データを FeatureSpace-Plot[ ] という関数でプロットすれば, 勝手に特徴量を判断しプロットしてくれます. 何も考えずに分類する方法です.

2 は花粉症もしくは逆に風邪である確率を, notebook1.pdf のように, Classify[ ] 関数を用いて, 計算し, 等高線を引けば, どの確率でどちらの判定をするかを決めれば, 境界線は引けます.

３は notebook1 のように, Classify[ ] 関数を用いて,

c = Classify[dataset, Method → "LogisticRegression"]

c[5.7, 5.0, "Probabilities"]

とすることにより, 与えられたデータの順に学習させると

⟨ |"風邪" → 0.480938, "花粉症" → 0.519062 | ⟩

という結果に成ります. つまり, どちらになるか分からないという結果に成ります. 「これはどうしようもないのか?なんとかならないか?」ということから, 4 の課題 に成ります.

4 は実は面倒なことをする必要はなく, データの学習する順番を変えるだけで, notebook2.pdf のように結果が劇的に変わります. 学習順序を変えるだけで,

⟨ |"風邪" → 0.162468, "花粉症" → 0.837532 | ⟩

という結果と成ります.

この例はデータの学習順序がニューラルネットワークの構築に強く影響するということです. この点, 注意して利用しなければなりません.

**課題 2** この場合は検査項目が非常に多いため, そのままデータを学習させると, notebook3.pdf の FeatureSpacePlot[ ] でプロットしてもラベルが正しく表示されません. 各データセットの判定はできますが, 正しい判定にはなっていません. この検査項目のうちどれかを選べば正しく判定できるのかを考えてもらいました. この問題正解はありません. そもそも, これらのデータで正しく判断できるのかどうかも分かりません.

このデータを多変量解析するときには, 各検査項目をそのままの数値で扱うのでなく, 値の範囲により, 3 段階から 5 段階に分類して行っています.

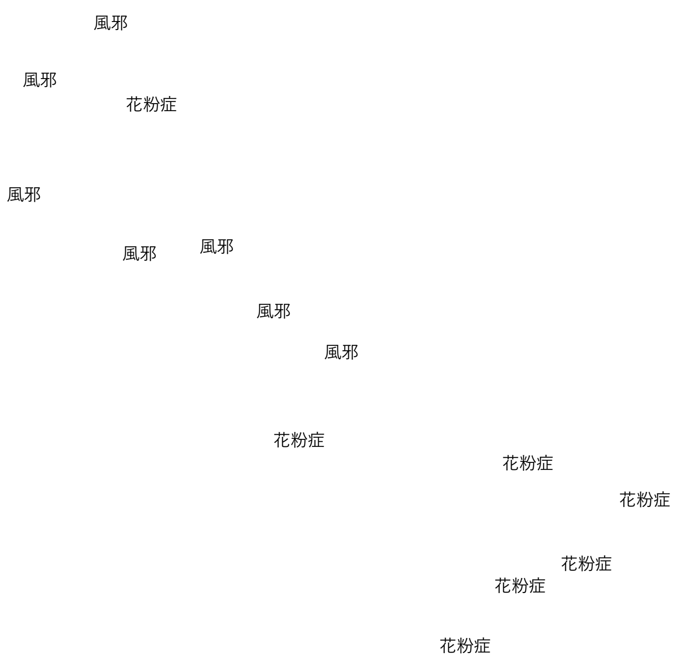
いろいろな工夫ができると思います. ただし, ニューラルネットの構造を工夫するのが全てではないということです.

## 講評

実行結果全てのハードコピーを提出した人, グループが非常に少なかったですね. 実際に, 自分たちがどのように実行し, どのような結果を得たのかを示すために, プログラムそのもの（この場合, Wolfram 言語 notebook）全体のハードコピーを付けましょう. 様々なニューラルネットで実行して, 分類を試みた解答が多くありました. そこまで要求しているわけではありませんでしたが.... また, 課題 2 については検査項目の組み合わせをいろいろ試して最もいいものを見つけだる作業になりますが, 実際に, それを実行していたものがありました. 着実な作業が重要に成ります.

様々な試みを試みていた解答を高く評価しました.

```
In[∘]:= dk = {{6.2, 2.3}, {9.9, 6.7}, {7.7, 3.2}, {6.9, 4.5}, {8.9, 4.3}, {4.3, 4.6}, {8.6, 2.0}}
Out[∘]= {{6.2, 2.3}, {9.9, 6.7}, {7.7, 3.2}, {6.9, 4.5}, {8.9, 4.3}, {4.3, 4.6}, {8.6, 2.}}

In[∘]:= dt = {{2.1, 6.0}, {0.9, 5.3}, {3.1, 3.2}, {4.1, 5.8}, {6.2, 1.3}, {4.9, 3.1}, {5.1, 1.8}}
Out[∘]= {{2.1, 6.}, {0.9, 5.3}, {3.1, 3.2}, {4.1, 5.8}, {6.2, 1.3}, {4.9, 3.1}, {5.1, 1.8}}

In[∘]:= dataset1 = Table[{dk〚i, 1〛, dk〚i, 2〛} → "花粉症", {i, 1, Length[dk]}]
Out[∘]= {{6.2, 2.3} → 花粉症, {9.9, 6.7} → 花粉症, {7.7, 3.2} → 花粉症,
  {6.9, 4.5} → 花粉症, {8.9, 4.3} → 花粉症, {4.3, 4.6} → 花粉症, {8.6, 2.} → 花粉症}

In[∘]:= dataset2 = Table[{dt〚i, 1〛, dt〚i, 2〛} → "風邪", {i, 1, Length[dt]}]
Out[∘]= {{2.1, 6.} → 風邪, {0.9, 5.3} → 風邪, {3.1, 3.2} → 風邪,
  {4.1, 5.8} → 風邪, {6.2, 1.3} → 風邪, {4.9, 3.1} → 風邪, {5.1, 1.8} → 風邪}

In[∘]:= dataset = Union[dataset1, dataset2]
Out[∘]= {{0.9, 5.3} → 風邪, {2.1, 6.} → 風邪, {3.1, 3.2} → 風邪,
  {4.1, 5.8} → 風邪, {4.3, 4.6} → 花粉症, {4.9, 3.1} → 風邪, {5.1, 1.8} → 風邪,
  {6.2, 1.3} → 風邪, {6.2, 2.3} → 花粉症, {6.9, 4.5} → 花粉症, {7.7, 3.2} → 花粉症,
  {8.6, 2.} → 花粉症, {8.9, 4.3} → 花粉症, {9.9, 6.7} → 花粉症}

In[∘]:= c = Classify[dataset, Method → "LogisticRegression"]
Out[∘]= ClassifierFunction[ Input type: NumericalVector (length: 2)
                            Classes: 風邪, 花粉症 ]

In[∘]:= FeatureSpacePlot[dataset]
Out[∘]=
```

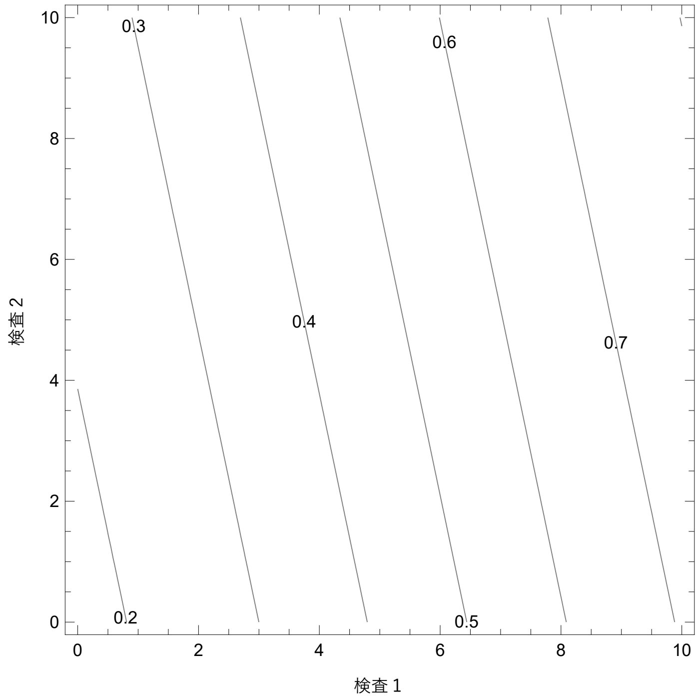



```
In[∘]:= c[{5.7, 5.0}]
Out[∘]= 花粉症
```

```
In[]:= c[{5.7, 5.0}, "Probabilities"]
Out[]= ⟨|風邪 → 0.480938, 花粉症 → 0.519062|⟩
```

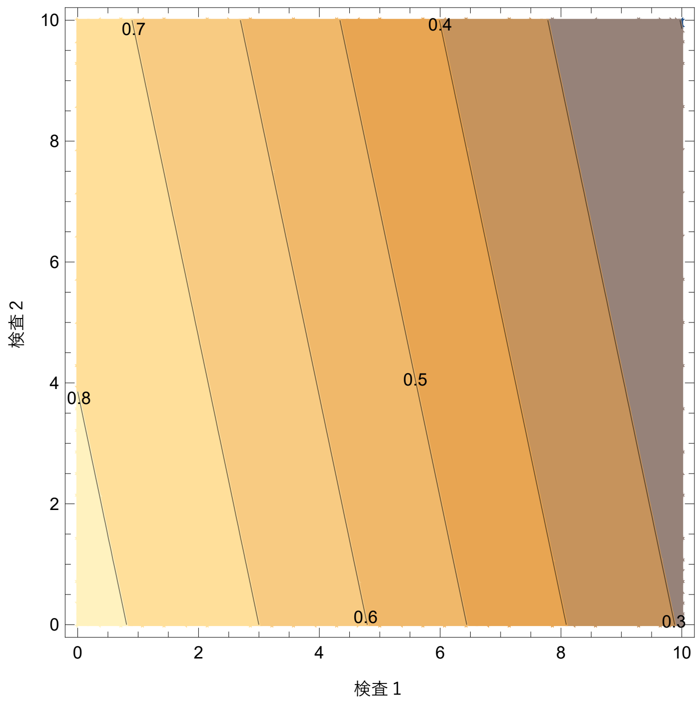
```
In[]:= ContourPlot[c[{x, y}, "Probability" → "花粉症"], {x, 0, 10}, {y, 0, 10},
  ContourLabels → True, FrameLabel → {"検査1", "検査2"}, ContourShading → False]
```

Out[]=



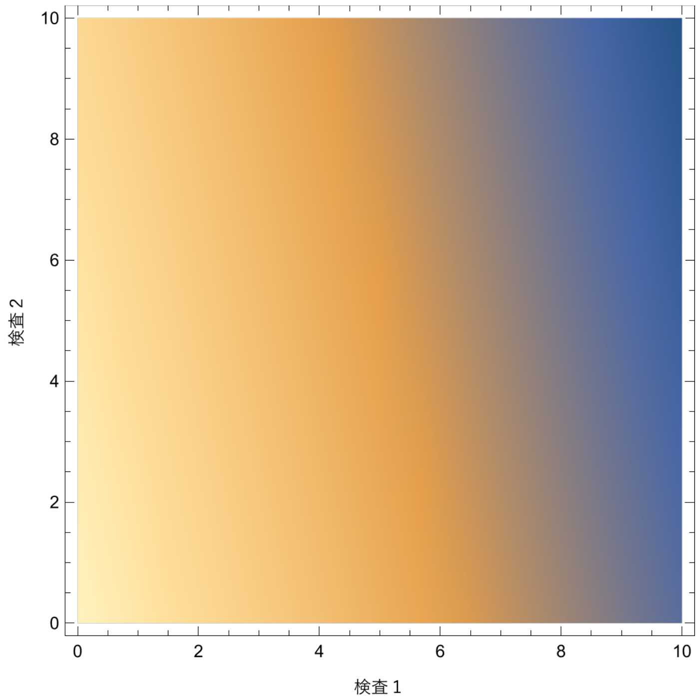

```
In[]:= ContourPlot[c[{x, y}, "Probability" → "風邪"], {x, 0, 10},
  {y, 0, 10}, ContourLabels → True, FrameLabel → {"検査1", "検査2"}]
```

Out[]=



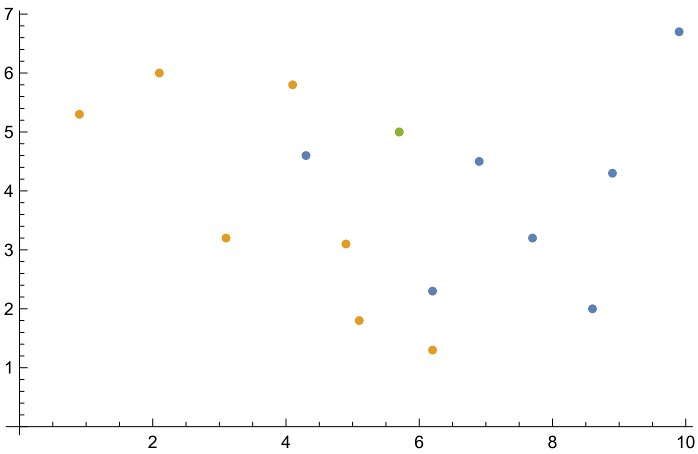
In[∘]:=
```
DensityPlot[c[{x, y}, "Probability" → "風邪"],
 {x, 0, 10}, {y, 0, 10}, FrameLabel → {"検査1", "検査2"}]
```

Out[∘]=



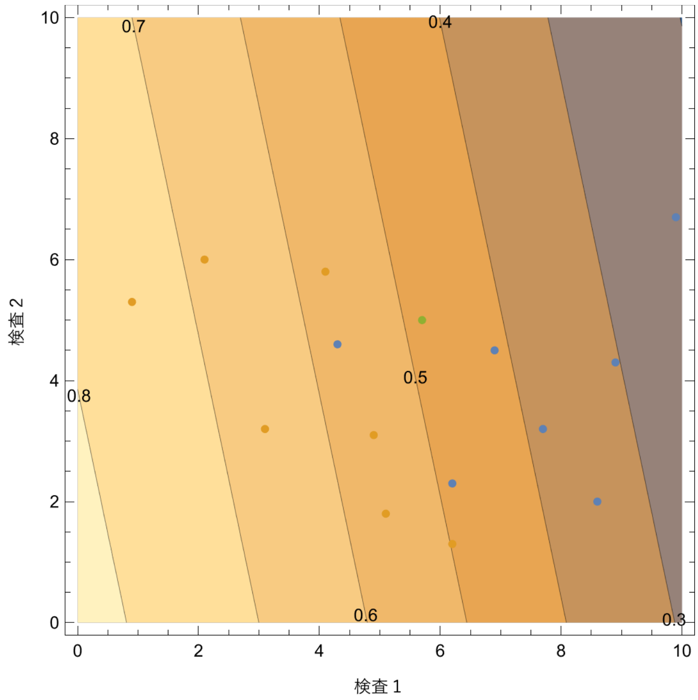

In[∘]:=
```
ListPlot[{dk, dt, {{5.7, 5.0}}}]
```

Out[∘]=

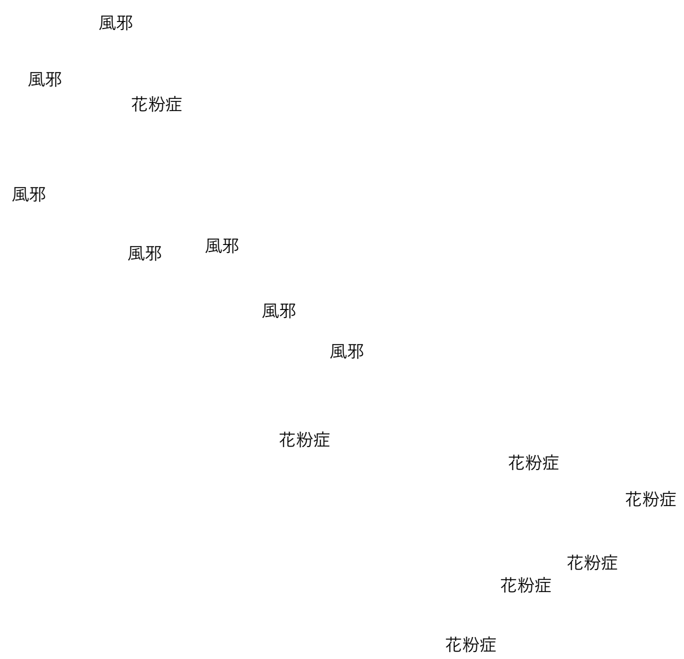


In[ ]:= 
```
Show[%%, %%%, %, FrameLabel → {"検査 1", "検査 2"}]
```

Out[ ]=



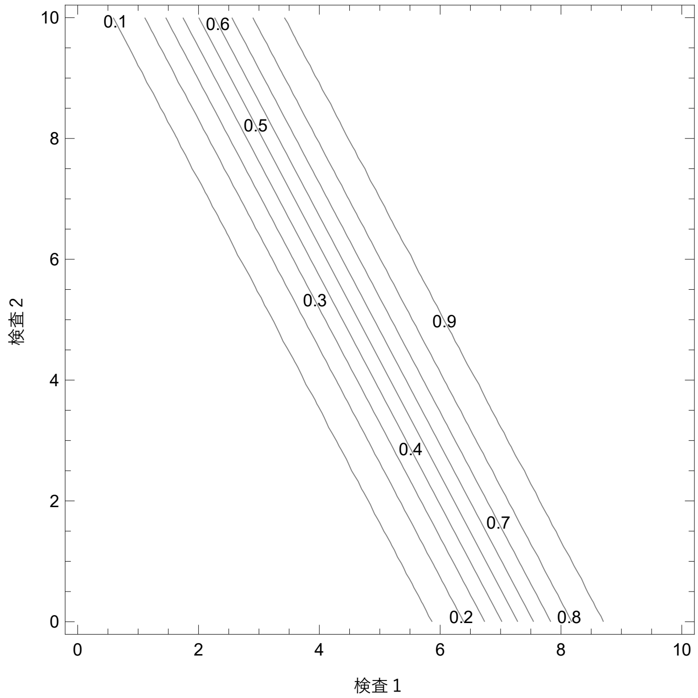
In[∘]:=
```
dk = RandomSample[
  {{6.2, 2.3}, {9.9, 6.7}, {7.7, 3.2}, {6.9, 4.5}, {8.9, 4.3}, {4.3, 4.6}, {8.6, 2.0}}]
```
Out[∘]=
```
{{8.6, 2.}, {9.9, 6.7}, {4.3, 4.6}, {6.9, 4.5}, {6.2, 2.3}, {7.7, 3.2}, {8.9, 4.3}}
```
In[∘]:=
```
dt = RandomSample[
  {{2.1, 6.0}, {0.9, 5.3}, {3.1, 3.2}, {4.1, 5.8}, {6.2, 1.3}, {4.9, 3.1}, {5.1, 1.8}}]
```
Out[∘]=
```
{{6.2, 1.3}, {4.1, 5.8}, {5.1, 1.8}, {2.1, 6.}, {3.1, 3.2}, {4.9, 3.1}, {0.9, 5.3}}
```
In[∘]:=
```
dataset1 = Table[{dk[[i, 1]], dk[[i, 2]]} → "花粉症", {i, 1, Length[dk]}]
```
Out[∘]=
```
{{8.6, 2.} → 花粉症, {9.9, 6.7} → 花粉症, {4.3, 4.6} → 花粉症,
 {6.9, 4.5} → 花粉症, {6.2, 2.3} → 花粉症, {7.7, 3.2} → 花粉症, {8.9, 4.3} → 花粉症}
```
In[∘]:=
```
dataset2 = Table[{dt[[i, 1]], dt[[i, 2]]} → "風邪", {i, 1, Length[dt]}]
```
Out[∘]=
```
{{6.2, 1.3} → 風邪, {4.1, 5.8} → 風邪, {5.1, 1.8} → 風邪,
 {2.1, 6.} → 風邪, {3.1, 3.2} → 風邪, {4.9, 3.1} → 風邪, {0.9, 5.3} → 風邪}
```
In[∘]:=
```
dataset = Union[dataset1, dataset2]
```
Out[∘]=
```
{{0.9, 5.3} → 風邪, {2.1, 6.} → 風邪, {3.1, 3.2} → 風邪,
 {4.1, 5.8} → 風邪, {4.3, 4.6} → 花粉症, {4.9, 3.1} → 風邪, {5.1, 1.8} → 風邪,
 {6.2, 1.3} → 風邪, {6.2, 2.3} → 花粉症, {6.9, 4.5} → 花粉症, {7.7, 3.2} → 花粉症,
 {8.6, 2.} → 花粉症, {8.9, 4.3} → 花粉症, {9.9, 6.7} → 花粉症}
```
In[∘]:=
```
c = Classify[dataset, Method → "LogisticRegression"]
```
Out[∘]=
```
ClassifierFunction[ Input type: NumericalVector (length: 2)
                    Classes: 風邪, 花粉症 ]
```
In[∘]:=
```
FeatureSpacePlot[dataset]
```
Out[∘]=



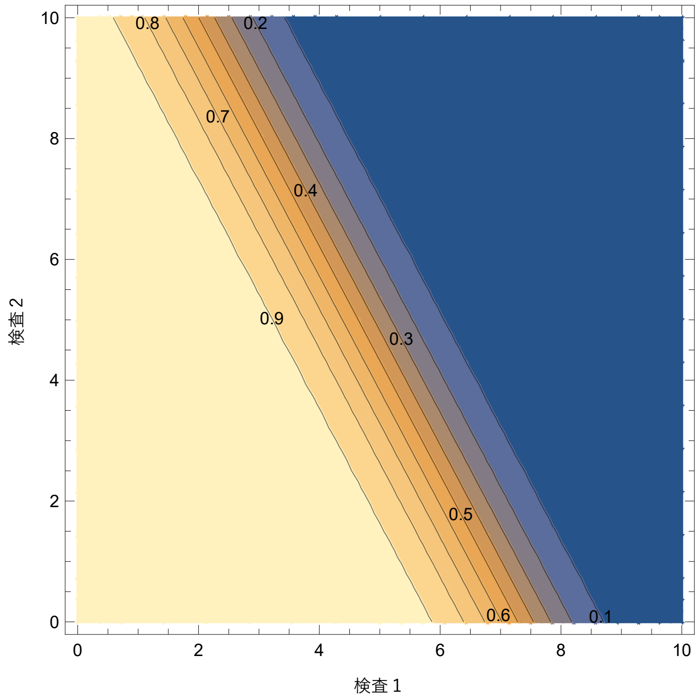
In[∘]:=
```
c[{5.7, 5.0}]
```
Out[∘]= 花粉症

In[∘]:=
```
c[{5.7, 5.0}, "Probabilities"]
```
Out[∘]= ⟨|風邪 → 0.162468, 花粉症 → 0.837532|⟩

In[∘]:=
```
ContourPlot[c[{x, y}, "Probability" → "花粉症"], {x, 0, 10}, {y, 0, 10},
 ContourLabels → True, FrameLabel → {"検査 1", "検査 2"}, ContourShading → False]
```
Out[∘]=

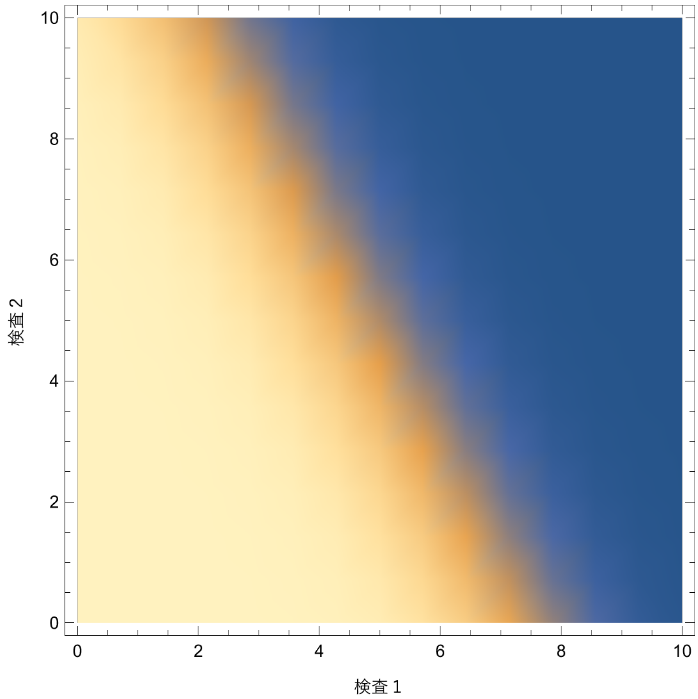



In[∘]:=
```
ContourPlot[c[{x, y}, "Probability" → "風邪"], {x, 0, 10},
 {y, 0, 10}, ContourLabels → True, FrameLabel → {"検査 1", "検査 2"}]
```
Out[∘]=



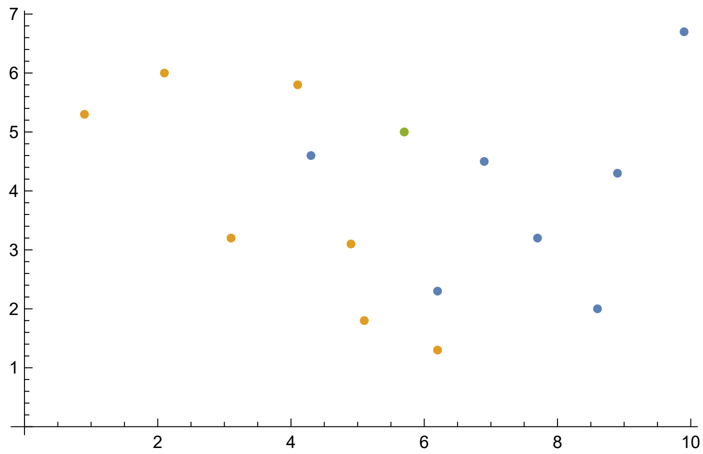
In[∘]:=
```
DensityPlot[c[{x, y}, "Probability" → "風邪"],
  {x, 0, 10}, {y, 0, 10}, FrameLabel → {"検査1", "検査2"}]
```

Out[∘]=



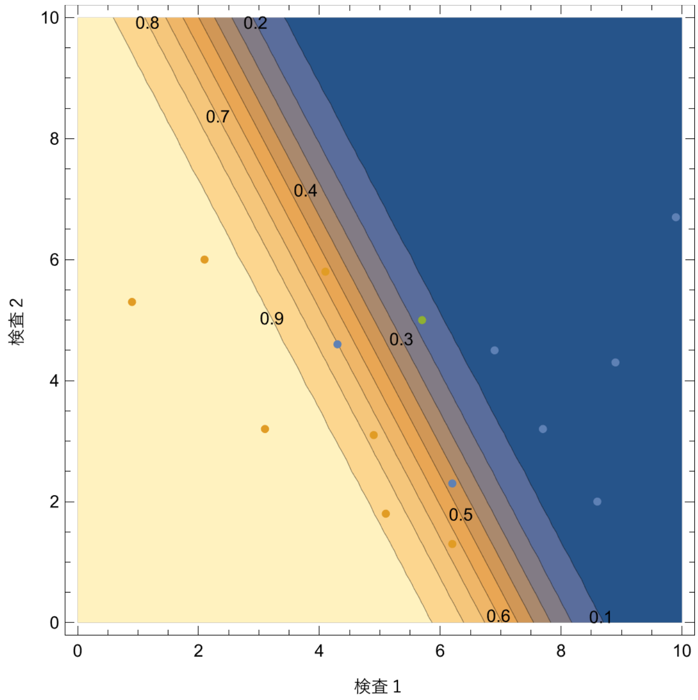

In[∘]:=
```
ListPlot[{dk, dt, {{5.7, 5.0}}}]
```

Out[∘]=



In[ ]:= `Show[%%, %%%, %, FrameLabel → {"検査1", "検査2"}]`

Out[ ]=



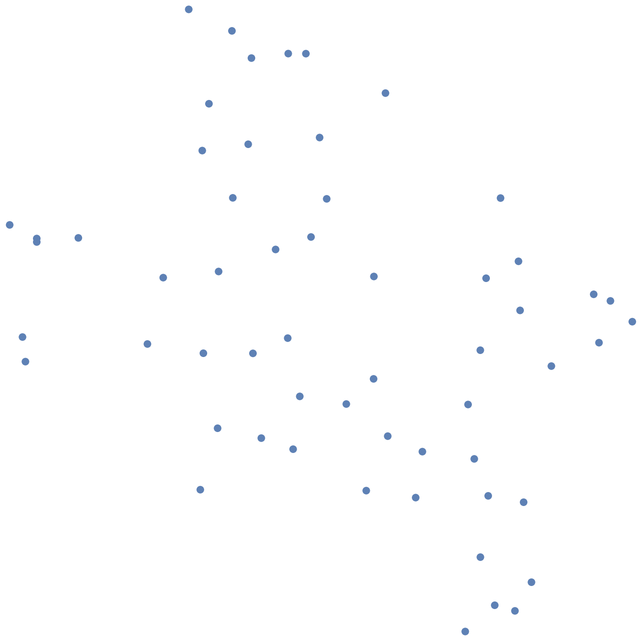
```
In[1]:= dn = {{3.3, 21.21, 54.55, 1.5, 30.30, 2, 1.44},
   {20, 35.00, 25.00, 0.0, 45.00, 6, 1.95}, {18, 27.78, 16.67, 0.0, 38.89, 3, 2.22},
   {31, 12.90, 32.26, 3.5, 51.61, 5, 1.52}, {28, 10.71, 50.00, 0.0, 46.43, 7, 1.23},
   {30, 6.67, 6.33, 3.0, 60.00, 5, 1.37}, {27, 3.70, 2.22, 2.0, 51.85, 1, 2.11},
   {33, 6.00, 4.85, 2.0, 57.58, 2, 1.32}, {57, 10.53, 31.58, 1.5, 56.14, 11, 1.55},
   {20, 15.00, 20.00, 2.0, 50.00, 8, 1.53}, {14, 14.29, 28.57, 4.0, 57.14, 4, 1.71},
   {36, 8.33, 36.11, 2.0, 66.67, 5, 1.42}, {30, 16.67, 36.67, 4.0, 30.00, 6, 1.42},
   {37, 10.81, 45.95, 4.0, 54.05, 7, 1.41}, {11, 45.46, 45.46, 0.0, 36.36, 6, 2.95},
   {17, 11.77, 35.29, 5.0, 58.82, 2, 1.59}, {34, 8.82, 32.35, 1.5, 55.88, 6, 1.34},
   {18, 22.22, 33.33, 0.0, 33.33, 4, 1.78}, {27, 22.22, 29.63, 0.0, 37.04, 5, 1.46}}

Out[•]= {{3.3, 21.21, 54.55, 1.5, 30.3, 2, 1.44},
  {20, 35., 25., 0., 45., 6, 1.95}, {18, 27.78, 16.67, 0., 38.89, 3, 2.22},
  {31, 12.9, 32.26, 3.5, 51.61, 5, 1.52}, {28, 10.71, 50., 0., 46.43, 7, 1.23},
  {30, 6.67, 6.33, 3., 60., 5, 1.37}, {27, 3.7, 2.22, 2., 51.85, 1, 2.11},
  {33, 6., 4.85, 2., 57.58, 2, 1.32}, {57, 10.53, 31.58, 1.5, 56.14, 11, 1.55},
  {20, 15., 20., 2., 50., 8, 1.53}, {14, 14.29, 28.57, 4., 57.14, 4, 1.71},
  {36, 8.33, 36.11, 2., 66.67, 5, 1.42}, {30, 16.67, 36.67, 4., 30., 6, 1.42},
  {37, 10.81, 45.95, 4., 54.05, 7, 1.41}, {11, 45.46, 45.46, 0., 36.36, 6, 2.95},
  {17, 11.77, 35.29, 5., 58.82, 2, 1.59}, {34, 8.82, 32.35, 1.5, 55.88, 6, 1.34},
  {18, 22.22, 33.33, 0., 33.33, 4, 1.78}, {27, 22.22, 29.63, 0., 37.04, 5, 1.46}}

In[2]:= dp = {{27, 14.82, 18.52, 7.0, 51.85, 7, 1.57},
   {47, 6.83, 59.57, 2.0, 61.70, 5, 1.15}, {19, 10.53, 5.26, 2.0, 68.42, 4, 1.16},
   {14, 35.71, 14.29, 0.0, 35.71, 3, 1.21}, {11, 26.67, 45.45, 0.0, 54.55, 3, 0.91},
   {15, 19.93, 33.33, 1.0, 60.00, 7, 1.13}, {26, 14.29, 11.54, 3.5, 46.15, 10, 1.33},
   {7, 16.00, 71.43, 0.0, 42.86, 3, 1.21}, {23, 16.67, 65.22, 0.0, 43.48, 4, 1.04},
   {18, 16.09, 50.00, 2.5, 61.11, 5, 1.36}, {11, 9.09, 27.27, 0.5, 18.18, 2, 0.64},
   {26, 11.54, 26.92, 6, 38.46, 6, 1.44}, {15, 13.33, 53.33, 1.5, 40.00, 4, 0.83},
   {19, 36.84, 21.05, 1.5, 36.84, 3, 1.45}, {19, 10.53, 78.95, 0.0, 73.68, 4, 1.13},
   {15, 6.67, 53.33, 0.0, 46.67, 5, 1.23}, {32, 6.25, 68.75, 0.0, 15.63, 5, 1.05},
   {33, 15.15, 15.15, 2.0, 60.61, 5, 0.92}, {42, 28.57, 23.81, 1.5, 35.71, 5, 1.48}}

Out[•]= {{27, 14.82, 18.52, 7., 51.85, 7, 1.57},
  {47, 6.83, 59.57, 2., 61.7, 5, 1.15}, {19, 10.53, 5.26, 2., 68.42, 4, 1.16},
  {14, 35.71, 14.29, 0., 35.71, 3, 1.21}, {11, 26.67, 45.45, 0., 54.55, 3, 0.91},
  {15, 19.93, 33.33, 1., 60., 7, 1.13}, {26, 14.29, 11.54, 3.5, 46.15, 10, 1.33},
  {7, 16., 71.43, 0., 42.86, 3, 1.21}, {23, 16.67, 65.22, 0., 43.48, 4, 1.04},
  {18, 16.09, 50., 2.5, 61.11, 5, 1.36}, {11, 9.09, 27.27, 0.5, 18.18, 2, 0.64},
  {26, 11.54, 26.92, 6, 38.46, 6, 1.44}, {15, 13.33, 53.33, 1.5, 40., 4, 0.83},
  {19, 36.84, 21.05, 1.5, 36.84, 3, 1.45}, {19, 10.53, 78.95, 0., 73.68, 4, 1.13},
  {15, 6.67, 53.33, 0., 46.67, 5, 1.23}, {32, 6.25, 68.75, 0., 15.63, 5, 1.05},
  {33, 15.15, 15.15, 2., 60.61, 5, 0.92}, {42, 28.57, 23.81, 1.5, 35.71, 5, 1.48}}
```

```
In[3]:= ds = {{11, 9.09, 18.19, 1.5, 54.55, 3, 1.50},
    {21, 0.00, 36.36, 3.0, 59.09, 4, 1.27}, {11, 27.27, 54.55, 0.0, 27.27, 6, 1.00},
    {13, 15.39, 23.08, 1.0, 53.85, 3, 1.38}, {16, 6.25, 43.75, 1.0, 93.75, 4, 0.94},
    {11, 0.0, 27.27, 1.0, 54.55, 1, 1.09}, {13, 0.0, 38.46, 3.0, 38.46, 1, 0.69},
    {29, 3.45, 41.38, 2.0, 51.72, 5, 0.50}, {17, 0.00, 64.71, 3.0, 64.71, 2, 0.80},
    {22, 0.00, 77.27, 1.0, 77.27, 1, 0.66}, {16, 6.25, 56.25, 0.0, 25.00, 1, 0.28},
    {12, 33.33, 58.33, 0.0, 50.00, 2, 1.12}, {18, 22.22, 44.44, 0.0, 55.56, 5, 1.60},
    {23, 0.00, 39.13, 1.0, 47.83, 4, 0.87}, {12, 0.00, 91.67, 0.0, 83.33, 4, 0.25},
    {7, 14.29, 57.14, 0.0, 85.71, 4, 1.09}, {44, 13.64, 59.09, 0.0, 40.91, 1, 0.41},
    {11, 18.18, 63.64, 1.0, 27.27, 2, 0.59}, {24, 12.50, 25.00, 1.0, 45.83, 3, 1.41}}

Out[•]= {{11, 9.09, 18.19, 1.5, 54.55, 3, 1.5},
    {21, 0., 36.36, 3., 59.09, 4, 1.27}, {11, 27.27, 54.55, 0., 27.27, 6, 1.},
    {13, 15.39, 23.08, 1., 53.85, 3, 1.38}, {16, 6.25, 43.75, 1., 93.75, 4, 0.94},
    {11, 0., 27.27, 1., 54.55, 1, 1.09}, {13, 0., 38.46, 3., 38.46, 1, 0.69},
    {29, 3.45, 41.38, 2., 51.72, 5, 0.5}, {17, 0., 64.71, 3., 64.71, 2, 0.8},
    {22, 0., 77.27, 1., 77.27, 1, 0.66}, {16, 6.25, 56.25, 0., 25., 1, 0.28},
    {12, 33.33, 58.33, 0., 50., 2, 1.12}, {18, 22.22, 44.44, 0., 55.56, 5, 1.6},
    {23, 0., 39.13, 1., 47.83, 4, 0.87}, {12, 0., 91.67, 0., 83.33, 4, 0.25},
    {7, 14.29, 57.14, 0., 85.71, 4, 1.09}, {44, 13.64, 59.09, 0., 40.91, 1, 0.41},
    {11, 18.18, 63.64, 1., 27.27, 2, 0.59}, {24, 12.5, 25., 1., 45.83, 3, 1.41}}

In[4]:= tdn = {24, 33.33, 25.00, 2.5, 41.67, 7, 2.19}

Out[•]= {24, 33.33, 25., 2.5, 41.67, 7, 2.19}

In[5]:= tdp = {12, 0.00, 91.67, 0.0, 75.00, 2, 1.00}

Out[•]= {12, 0., 91.67, 0., 75., 2, 1.}

In[6]:= tds = {24, 12.50, 25.00, 1.0, 45.83, 3, 1.42}

Out[•]= {24, 12.5, 25., 1., 45.83, 3, 1.42}

In[7]:= dataset1 = Table[dn⟦i⟧ → "n", {i, 1, Length[dn]}]

Out[•]= {{3.3, 21.21, 54.55, 1.5, 30.3, 2, 1.44} → n,
    {20, 35., 25., 0., 45., 6, 1.95} → n, {18, 27.78, 16.67, 0., 38.89, 3, 2.22} → n,
    {31, 12.9, 32.26, 3.5, 51.61, 5, 1.52} → n, {28, 10.71, 50., 0., 46.43, 7, 1.23} → n,
    {30, 6.67, 6.33, 3., 60., 5, 1.37} → n, {27, 3.7, 2.22, 2., 51.85, 1, 2.11} → n,
    {33, 6., 4.85, 2., 57.58, 2, 1.32} → n, {57, 10.53, 31.58, 1.5, 56.14, 11, 1.55} → n,
    {20, 15., 20., 2., 50., 8, 1.53} → n, {14, 14.29, 28.57, 4., 57.14, 4, 1.71} → n,
    {36, 8.33, 36.11, 2., 66.67, 5, 1.42} → n, {30, 16.67, 36.67, 4., 30., 6, 1.42} → n,
    {37, 10.81, 45.95, 4., 54.05, 7, 1.41} → n, {11, 45.46, 45.46, 0., 36.36, 6, 2.95} → n,
    {17, 11.77, 35.29, 5., 58.82, 2, 1.59} → n, {34, 8.82, 32.35, 1.5, 55.88, 6, 1.34} → n,
    {18, 22.22, 33.33, 0., 33.33, 4, 1.78} → n, {27, 22.22, 29.63, 0., 37.04, 5, 1.46} → n}
```

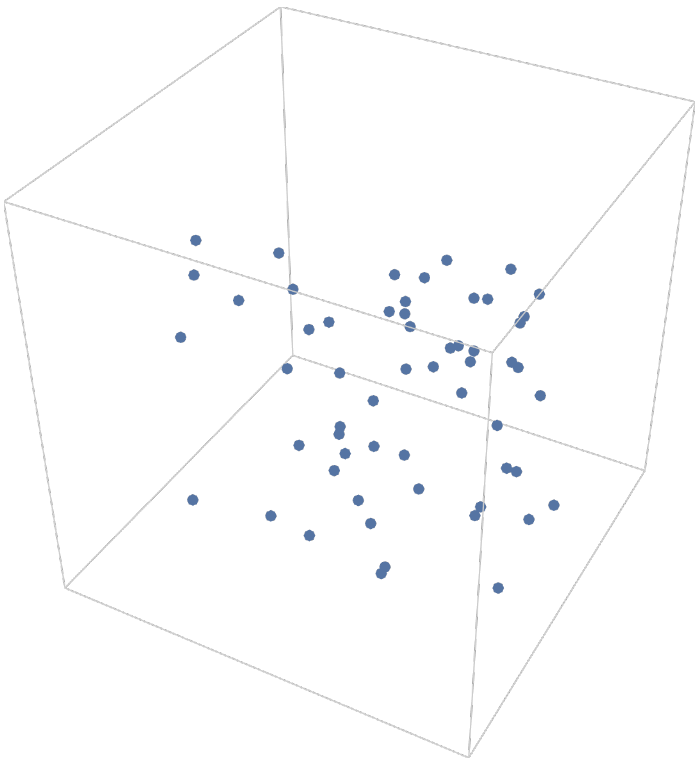
```
In[8]:= dataset2 = Table[dp[[i]] → "p", {i, 1, Length[dp]}]

Out[•]= {{27, 14.82, 18.52, 7., 51.85, 7, 1.57} → p,
  {47, 6.83, 59.57, 2., 61.7, 5, 1.15} → p, {19, 10.53, 5.26, 2., 68.42, 4, 1.16} → p,
  {14, 35.71, 14.29, 0., 35.71, 3, 1.21} → p, {11, 26.67, 45.45, 0., 54.55, 3, 0.91} → p,
  {15, 19.93, 33.33, 1., 60., 7, 1.13} → p, {26, 14.29, 11.54, 3.5, 46.15, 10, 1.33} → p,
  {7, 16., 71.43, 0., 42.86, 3, 1.21} → p, {23, 16.67, 65.22, 0., 43.48, 4, 1.04} → p,
  {18, 16.09, 50., 2.5, 61.11, 5, 1.36} → p, {11, 9.09, 27.27, 0.5, 18.18, 2, 0.64} → p,
  {26, 11.54, 26.92, 6, 38.46, 6, 1.44} → p, {15, 13.33, 53.33, 1.5, 40., 4, 0.83} → p,
  {19, 36.84, 21.05, 1.5, 36.84, 3, 1.45} → p, {19, 10.53, 78.95, 0., 73.68, 4, 1.13} → p,
  {15, 6.67, 53.33, 0., 46.67, 5, 1.23} → p, {32, 6.25, 68.75, 0., 15.63, 5, 1.05} → p,
  {33, 15.15, 15.15, 2., 60.61, 5, 0.92} → p, {42, 28.57, 23.81, 1.5, 35.71, 5, 1.48} → p}

In[9]:= dataset3 = Table[ds[[i]] → "s", {i, 1, Length[ds]}]

Out[•]= {{11, 9.09, 18.19, 1.5, 54.55, 3, 1.5} → s,
  {21, 0., 36.36, 3., 59.09, 4, 1.27} → s, {11, 27.27, 54.55, 0., 27.27, 6, 1.} → s,
  {13, 15.39, 23.08, 1., 53.85, 3, 1.38} → s, {16, 6.25, 43.75, 1., 93.75, 4, 0.94} → s,
  {11, 0., 27.27, 1., 54.55, 1, 1.09} → s, {13, 0., 38.46, 3., 38.46, 1, 0.69} → s,
  {29, 3.45, 41.38, 2., 51.72, 5, 0.5} → s, {17, 0., 64.71, 3., 64.71, 2, 0.8} → s,
  {22, 0., 77.27, 1., 77.27, 1, 0.66} → s, {16, 6.25, 56.25, 0., 25., 1, 0.28} → s,
  {12, 33.33, 58.33, 0., 50., 2, 1.12} → s, {18, 22.22, 44.44, 0., 55.56, 5, 1.6} → s,
  {23, 0., 39.13, 1., 47.83, 4, 0.87} → s, {12, 0., 91.67, 0., 83.33, 4, 0.25} → s,
  {7, 14.29, 57.14, 0., 85.71, 4, 1.09} → s, {44, 13.64, 59.09, 0., 40.91, 1, 0.41} → s,
  {11, 18.18, 63.64, 1., 27.27, 2, 0.59} → s, {24, 12.5, 25., 1., 45.83, 3, 1.41} → s}
```

```
In[13]:= dataset = Union[dataset1, dataset2, dataset3]

Out[•]= {{3.3, 21.21, 54.55, 1.5, 30.3, 2, 1.44} → n,
  {7, 14.29, 57.14, 0., 85.71, 4, 1.09} → s, {7, 16., 71.43, 0., 42.86, 3, 1.21} → p,
  {11, 0., 27.27, 1., 54.55, 1, 1.09} → s, {11, 9.09, 18.19, 1.5, 54.55, 3, 1.5} → s,
  {11, 9.09, 27.27, 0.5, 18.18, 2, 0.64} → p, {11, 18.18, 63.64, 1., 27.27, 2, 0.59} → s,
  {11, 26.67, 45.45, 0., 54.55, 3, 0.91} → p, {11, 27.27, 54.55, 0., 27.27, 6, 1.} → s,
  {11, 45.46, 45.46, 0., 36.36, 6, 2.95} → n, {12, 0., 91.67, 0., 83.33, 4, 0.25} → s,
  {12, 33.33, 58.33, 0., 50., 2, 1.12} → s, {13, 0., 38.46, 3., 38.46, 1, 0.69} → s,
  {13, 15.39, 23.08, 1., 53.85, 3, 1.38} → s, {14, 14.29, 28.57, 4., 57.14, 4, 1.71} → n,
  {14, 35.71, 14.29, 0., 35.71, 3, 1.21} → p, {15, 6.67, 53.33, 0., 46.67, 5, 1.23} → p,
  {15, 13.33, 53.33, 1.5, 40., 4, 0.83} → p, {15, 19.93, 33.33, 1., 60., 7, 1.13} → p,
  {16, 6.25, 43.75, 1., 93.75, 4, 0.94} → s, {16, 6.25, 56.25, 0., 25., 1, 0.28} → s,
  {17, 0., 64.71, 3., 64.71, 2, 0.8} → s, {17, 11.77, 35.29, 5., 58.82, 2, 1.59} → n,
  {18, 16.09, 50., 2.5, 61.11, 5, 1.36} → p, {18, 22.22, 33.33, 0., 33.33, 4, 1.78} → n,
  {18, 22.22, 44.44, 0., 55.56, 5, 1.6} → s, {18, 27.78, 16.67, 0., 38.89, 3, 2.22} → n,
  {19, 10.53, 5.26, 2., 68.42, 4, 1.16} → p, {19, 10.53, 78.95, 0., 73.68, 4, 1.13} → p,
  {19, 36.84, 21.05, 1.5, 36.84, 3, 1.45} → p, {20, 15., 20., 2., 50., 8, 1.53} → n,
  {20, 35., 25., 0., 45., 6, 1.95} → n, {21, 0., 36.36, 3., 59.09, 4, 1.27} → s,
  {22, 0., 77.27, 1., 77.27, 1, 0.66} → s, {23, 0., 39.13, 1., 47.83, 4, 0.87} → s,
  {23, 16.67, 65.22, 0., 43.48, 4, 1.04} → p, {24, 12.5, 25., 1., 45.83, 3, 1.41} → s,
  {26, 11.54, 26.92, 6, 38.46, 6, 1.44} → p, {26, 14.29, 11.54, 3.5, 46.15, 10, 1.33} → p,
  {27, 3.7, 2.22, 2., 51.85, 1, 2.11} → n, {27, 14.82, 18.52, 7., 51.85, 7, 1.57} → p,
  {27, 22.22, 29.63, 0., 37.04, 5, 1.46} → n, {28, 10.71, 50., 0., 46.43, 7, 1.23} → n,
  {29, 3.45, 41.38, 2., 51.72, 5, 0.5} → s, {30, 6.67, 6.33, 3., 60., 5, 1.37} → n,
  {30, 16.67, 36.67, 4., 30., 6, 1.42} → n, {31, 12.9, 32.26, 3.5, 51.61, 5, 1.52} → n,
  {32, 6.25, 68.75, 0., 15.63, 5, 1.05} → p, {33, 6., 4.85, 2., 57.58, 2, 1.32} → n,
  {33, 15.15, 15.15, 2., 60.61, 5, 0.92} → p, {34, 8.82, 32.35, 1.5, 55.88, 6, 1.34} → n,
  {36, 8.33, 36.11, 2., 66.67, 5, 1.42} → n, {37, 10.81, 45.95, 4., 54.05, 7, 1.41} → n,
  {42, 28.57, 23.81, 1.5, 35.71, 5, 1.48} → p, {44, 13.64, 59.09, 0., 40.91, 1, 0.41} → s,
  {47, 6.83, 59.57, 2., 61.7, 5, 1.15} → p, {57, 10.53, 31.58, 1.5, 56.14, 11, 1.55} → n}

In[14]:= c = Classify[dataset, Method → "LogisticRegression"]

Out[•]= ClassifierFunction[ Input type: Mixed (number: 7)
                            Classes: n, p, s ]
```

In[15]:= **FeatureSpacePlot[dataset]**

Out[◦]=


In[16]:= **FeatureSpacePlot3D[dataset]**

Out[◦]=


In[17]:= **c[tdn]**

Out[◦]= n

In[18]:= **c[tdn, "Probabilities"]**

Out[◦]= <|n → 0.736944, p → 0.238183, s → 0.0248727|>

```
In[19]:= c[tdp]
Out[]= s
In[20]:= c[tdp, "Probabilities"]
Out[]= <|n → 0.0509673, p → 0.105749, s → 0.843284|>
In[21]:= c[tds]
Out[]= n
In[22]:= c[tds, "Probabilities"]
Out[]= <|n → 0.425633, p → 0.276019, s → 0.298348|>
```

第 26 回数理科学コンクール課題解説:追加資料

## 課題 2

単気筒機関の運動を Jacob Pieter Den Hartog:Mechanical Vibrations(1934) に従って解析してみましょう.

気筒の構成要素は, ピストン, コンロッド (連結棒), クランクです. ピストンが上下運動をし, 上下運動 (直線的運動) を連結棒がクランクに伝えて回転運動に変換します.

以下のように

$$
\begin{aligned}
x_p &= \text{ピストンの上の死点から下方へ向かう変位} \\
\omega t &= \text{回転系の上の死点からのクランク角} \\
r &= \text{クランク半径} \\
\ell &= \text{コンロッドの長さ}
\end{aligned}
$$

記号を決めると, ピストンの上の死点から下方へ向かう変位 $x_p$, 速度 $\dot{x}_p$, 加速度 $\ddot{x}_p$ は, 値の無視できる項を消去すると, $x_p$ の精密解から導かれる $\dot{x}_p$, $\ddot{x}_p$ は

$$
\begin{aligned}
x_p &= \left(r + \frac{r^2}{4\ell}\right) - r\left(\cos\omega t + \frac{r}{4\ell}\cos 2\omega t\right) \\
\dot{x}_p &= r\omega\left(\cos\omega t + \frac{r}{2\ell}\cos 2\omega t\right) \\
\ddot{x}_p &= r\omega\left(\cos\omega t + \frac{r}{\ell}\cos 2\omega t\right)
\end{aligned}
$$

と成ります. $\ddot{x}_p$ にピストンの質量を掛けると, 慣性力が計算できます. 慣性力に周期に関する 1 次, 2 次の項が発生します

次に, ピストン系の運動をコンロッドの頂点にピストン系の全質量 $m_c$ がある質点運動で近似して解析します. $m_c$ の位置を $(x_c, y_c)^\top$ とすれば, 速度 $(\dot{x}_c, \dot{y}_c)^\top$, 加速度 $(\ddot{x}_c, \ddot{y}_c)^\top$ は, それぞれ,

$$
\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \begin{pmatrix} r(1-\cos\omega t) \\ -r\sin\omega t \end{pmatrix} \tag{1}
$$

$$
\begin{pmatrix} \dot{x}_c \\ \dot{y}_c \end{pmatrix} = \begin{pmatrix} r\omega\cos\omega t \\ -r\omega\sin\omega t \end{pmatrix} \tag{2}
$$

$$
\begin{pmatrix} \ddot{x}_c \\ \ddot{y}_c \end{pmatrix} = \begin{pmatrix} r\omega^2\cos\omega t \\ r\omega^2\sin\omega t \end{pmatrix} \tag{3}
$$

と成ります.

$m_c$ を回転成分 $m_r$, 水平運動成分 $m_p$ に分けて考えると, 水平慣性力 $X$, 垂直慣性力 $Y$, モーメント $M$ は, それぞれ,

$$
X = (m_r + m_p)r\omega^2\cos\omega t + m_p\frac{r^2\omega^2}{\ell}\cos 2\omega t \tag{4}
$$

$$
\begin{aligned}
Y &= m_r r\omega \sin\omega t && (5)\\
M &= \frac{1}{2} m_p r^2 \omega^2 \left(\frac{r}{2\ell}\sin\omega t - \sin 2\omega t - \frac{3r}{2\ell}\sin\omega t\right) && (6)
\end{aligned}
$$

と成ります. $m_r$ は回転質量成分なので, トルクを発生させません.

$n$ 気筒機関の 1 番クランクから $n$ 番クランクまでの軸上の距離を $L_n$ とします. また, 1 番クランクの回転角から $n$ 番クランクの回転角までの角度 (クランク角) を $\alpha_n$ とします. そして, $n$ 番クランクは, 1 クランクより $\alpha_n$ 進んで回転しているものとします.

従って, $n$ 番クランクの位置での 1 次慣性力の方向は $\alpha_n$, 2 次慣性力の方向は $2 \times \alpha_n$, それぞれ, 1 番クランクの回転角より進んでいます. $n$ 番ピストン形の慣性力は, モーメントにクランク間距離 $L_n$ を乗じたもの

$$
F_n = L_n \times \frac{1}{2} m_p r^2 \omega^2 \left(\frac{r}{2\ell}\sin(\omega t - \alpha_n) - \sin 2(\omega t - \alpha_n) - \frac{3r}{2\ell}\sin 3(\omega t - \alpha_n)\right) \tag{7}
$$

です. これらの関係を平面ベクトルに表し, その釣合を調べると, 何が打消されるかが分かります. その関係を下の表にまとめます.

表 1: 死点の位相角

| 気筒数 | 位相角 | | | | | | | | 釣合 |
|---|---|---|---|---|---|---|---|---|---|
| 4 | 0 | 90 | 270 | 180 | | | | | 1 次,2 次慣性力 |
| | | | | | | | | | 2 次モーメント |
| 4 | 0 | 180 | 180 | 0 | | | | | 1 次慣性力 |
| | | | | | | | | | 1 次モーメント |
| 6 | 0 | 120 | 240 | 240 | 120 | 0 | | | 1 次,2 次慣性力 |
| | | | | | | | | | 1 次, 2 次モーメント |
| 8 | 0 | 180 | 90 | 270 | 270 | 90 | 180 | 0 | 完全 |